



FAKULTA APLIKOVANÝCH VĚD
ZÁPADOČESKÉ UNIVERZITY
V PLZNI

KATEDRA INFORMATIKY
A VÝPOČETNÍ TECHNIKY

Semestrální práce

Souborový systém založený na i-uzlech

Jakub Vokoun

Obsah

1	Zadání	2
2	Analýza úlohy	3
3	Popis implementace	4
4	Uživatelská příručka	5
5	Závěr	6
A	Zadání	7

Zadání

1

Na straně: 7.

Analýza úlohy

2

Popis implementace

3

Uživatelská příručka

4

Závěr

5

Zadání

A

Následující strany obasahují kompletní a nezkrácené zadání semestrální práce.

Semestrální práce ZOS 2025 (verze dokumentu 01)

Tématem semestrální práce bude práce se zjednodušeným souborovým systémem založeným na **i-uzlech**. Vaším cílem bude splnit několik vybraných úloh.

Základní funkčnost, kterou musí program splňovat. Formát výpisů je závazný, můžete si však přidat další pomocná hlášení, jakou jsou podrobnosti k chybám či doplňující vypisované informace.

Program bude mít **jeden parametr** a tím bude **název** vašeho souborového systému. Po spuštění bude program čekat na zadání jednotlivých příkazů s minimální funkčností viz níže (všechny soubory mohou být zadány jak absolutní, tak relativní cestou):

- 1) Zkopíruje soubor s1 do umístění s2

```
cp s1 s2
```

Možný výsledek:

```
OK  
FILE NOT FOUND (není zdroj)  
PATH NOT FOUND (neexistuje cílová cesta)
```

- 2) Přesune soubor s1 do umístění s2, nebo přejmenuje s1 na s2

```
mv s1 s2
```

Možný výsledek:

```
OK  
FILE NOT FOUND (není zdroj)  
PATH NOT FOUND (neexistuje cílová cesta)
```

- 3) Smaže soubor s1

```
rm s1
```

Možný výsledek:

```
OK  
FILE NOT FOUND
```

- 4) Vytvoří adresář a1

```
mkdir a1
```

Možný výsledek:

```
OK  
PATH NOT FOUND (neexistuje zadaná cesta)  
EXIST (nelze založit, již existuje)
```

5) Smaže prázdný adresář a1

```
rmdir a1
```

Možný výsledek:

OK

FILE NOT FOUND (neexistující adresář)

NOT EMPTY (adresář obsahuje podadresáče, nebo soubory)

6) Vypíše obsah adresáře a1

```
ls a1  
ls
```

Možný výsledek:

FILE: f1
DIR: a2

PATH NOT FOUND (neexistující adresář)

7) Vypíše obsah textového souboru s1 na obrazovku

```
cat s1
```

Možný výsledek:

OBSAH TEXTOVÉHO SOUBORU

FILE NOT FOUND (není zdroj)

8) Změní aktuální cestu do adresáře a1

```
cd a1
```

Možný výsledek:

OK

PATH NOT FOUND (neexistující cesta)

9) Vypíše aktuální cestu

```
pwd
```

Možný výsledek:

PATH

10) Vypíše informace o souboru/adresáři s1/a1 (v jakých clusterech se nachází)

```
info a1  
info s1
```

Možný výsledek:

```
NAME - SIZE - i-node NUMBER - přímé a nepřímé odkazy  
FILE NOT FOUND (není zdroj)  
Může vypadat např. takto u clusteru velikost 1024B:  
ahoj.txt - 1800 B - i-uzel 7 - přímé odkazy 101, 102
```

11) Nahraje soubor s1 z pevného disku do umístění s2 ve vašem FS

```
incp s1 s2
```

Možný výsledek:

```
OK  
FILE NOT FOUND (není zdroj)  
PATH NOT FOUND (neexistuje cílová cesta)
```

12) Nahraje soubor s1 z vašeho FS do umístění s2 na pevném disku

```
outcp s1 s2
```

Možný výsledek:

```
OK  
FILE NOT FOUND (není zdroj)  
PATH NOT FOUND (neexistuje cílová cesta)
```

13) Načte soubor z pevného disku, ve kterém budou jednotlivé příkazy, a začne je sekvenčně vykonávat. Formát je 1 příkaz/1řádek

```
load s1
```

Možný výsledek:

```
OK  
FILE NOT FOUND (není zdroj)
```

14) Příkaz provede formát souboru, který byl zadán jako parametr při spuštění programu na souborový systém dané velikosti. Pokud už soubor nějaká data obsahoval, budou přemazána. Pokud soubor neexistoval, bude vytvořen.

```
format 600MB  
Možný výsledek:  
OK  
CANNOT CREATE FILE
```

- 15) Příkaz exit – ukončení práce s vaším programem.
- 16) Příkaz statfs – vypíše statistiky souborového systému, jako je velikost, počet obsazených a volných bloků, počet obsazených a volných i-uzlů, počet adresářů.

Budeme předpokládat korektní zadání syntaxe příkazů, nikoliv však sémantiky (tj. např. cp s1 zadáno nebude, ale může být zadáno cat s1, kde s1 neexistuje).

Informace k zadání a omezením

- Maximální délka názvu souboru bude $8+3=11$ znaků (jméno.přípona) + \0 (ukončovací znak v C/C++), tedy 12 bytů.
- Každý název bude zabírat právě 12 bytů (do délky 12 bytů doplníte \0 - při kratších názvech).

Nad vytvořeným a naplněným souborovým systémem umožněte provedení následujících operací:

- Hardlink (příkaz *ln s1 s2*) – pokud login studenta začíná **a-i**
Vytvoří hard link na soubor s1 s názvem s2. Dále se s ním pracuje očekávaným způsobem, tedy např. cat s2 vypíše stejný obsah jako cat s1. Také dojde k úpravě příkazu info, aby zobrazoval i počet hardlinků odkazujících na daná data.
- Manipulace se soubory – pokud login studenta začíná **j-r**
příkaz *xcp s1 s2 s3* - Vytvoří soubor s3, který bude spojením souborů s1 a s2.
příkaz *add s1 s2* – Přidá na konec souboru s1 obsah souboru s2
- Symbolický link (*slink s1 s2*) – pokud login studenta začíná **s-z**
Vytvoří symbolický link na soubor s1 s názvem s2. Dále se s ním pracuje očekávaným způsobem, tedy např. cat s2 vypíše obsah souboru s1. Také dojde k úpravě příkazů ls a info, kde uvidíme, že daný soubor je symbolickým linkem.

Odevzdání práce

Práci včetně dokumentace pošlete svému cvičícímu e-mailem. V případě velkého objemu dat můžete využít různé služby pro přenos souborů (Cesnet FileSender, uschovna.cz).

Osobní předvedení práce cvičícímu. Referenčním strojem je školní PC v UC326 nebo UC329. Práci můžete ukázat i na svém notebooku. Konkrétní datum a čas předvedení práce si domluvte e-mailem se cvičícím, sdělí vám časová okna, kdy můžete práci ukázat.

Do kdy musím semestrální práci odevzdat?

- Zápočet musíte získat do mezního data pro získání zápočtu (10. února 2026).
- A samozřejmě je třeba mít zápočet dříve, než půjdete na zkoušku (alespoň 1 den předem).

Hodnocení

Při kontrole semestrální práce bude hodnocena:

- Kvalita a čitelnost kódu včetně komentářů
- Funkčnost a kvalita řešení
- Kvalita dokumentace

