

# [SOI] Lab 6 - Dysk wirtualny

Szczawiński Kamil

10 stycznia 2025

## 1 Zadanie

Celem zadania jest realizacja systemu plików z jednopoziomowym katalogiem. Elementem katalogu plików jest opis pliku, zawierający co najmniej nazwę, wielkość i sposób rozmieszczenia pliku na wirtualnym dysku. Ponadto, od dysku wymaga się, by obsługiwał:

- pliki ukryte (dot files)
- nieciągłą alokację plików
- nazwy plików o dowolnej długości
- nazwy plików zawierające spacje

Dysk będzie obsługiwany przez zewnętrzny program, wspierający następujące operacje:

- usuwanie dysku
- tworzenie dysku
- usuwanie plików z dysku
- kopiowanie plików z systemu Minix
- kopiowanie plików do systemu Minix
- wyświetlanie katalogu (opcjonalnie plików ukrytych)

## 2 Struktura

Dysk jest przedstawiony w formie pliku binarnego, składający się z pięciu części:

1. nagłówek
2. bitmapa i-node'ów
3. tabela i-node'ów
4. bitmapa bloków danych
5. bloki danych

Strukturę poszczególnych elementów opisałem w dalszej części dokumentu.

### 2.1 Nagłówek / Superblok

- **Rozmiar dysku: 4 bajty** - kluczowa informacja potrzebna dla ochrony przed przepełnieniem. 32 bity pozwalają na rozmiar do ok. 4 GB który powinien całkowicie wystarczyć na potrzeby ćwiczenia.
- **Rozmiar bloku danych: 4 bajty**
- **Maksymalna liczba plików: 4 bajty**
- **Offset tabeli plików: 4 bajty**
- **Offset bloków danych: 4 bajty**

### 2.2 Bitmapa zajętych i-node'ów

Bitmapa o długości równej liczbie i-node'ów. Wartości bitów równe 0 lub 1 w zależności czy i'ty z kolei i-node jest wykorzystywany przez plik. Zastosowanie: Na jej podstawie możemy obliczać adresy zajętych w tabeli i-node'ów.

### 2.3 Tabela plików / i-node'ów

Rozmiar tabeli będzie stały, wyliczony na podstawie danych odczytanych z nagłówka.

Taka decyzja projektowa, ogranicza nas pod względem maksymalnej ilości plików przechowywanych na dysku, jednak umożliwia sprawne odnajdywanie adresów/offsetów interesujących nas bloków danych.

Struktura pojedynczego wpisu:

- **Flaga ukrycia: 4 bajty** - wykorzystywana przy listowaniu plików
- **Nazwa pliku 1: 64 bajty**
- **Nazwa pliku 2: 4 bajty** - w przypadku gdyby rozmiar nazwy przekraczał 64 bajty, pole to będzie wskazywało na standardowy blok danych zawierający dalszą część nazwy.
- **Rozmiar pliku: 4 bajty**
- **Adres pierwszego bloku danych pliku: 4 bajty**

Rozmiar - 80 bajtów

## 2.4 Bitmapa zajętych bloków danych

Bitmapa o długości równej liczbie bloków danych. Wartości bitów równe 0 lub 1 w zależności czy i'ty z kolei blok jest zajęty.

Zastosowanie: Zawiera informacje o zajętej przestrzeni oraz dziurach w blokach danych.

## 2.5 Bloki danych

Pliki podzielone będą na bloki o rozmiarze zdefiniowanym w nagłówku.

- **Dane**
- **Adres następnego bloku: 4 bajty**