

オペレーティングシステム 最終レポート課題

担当教員:村田 正幸

提出者名:小林 亮太

学籍番号:09B19025

所属:ソフトウェア科学コース

提出年月日:令和3年6月6日

提出締切日:令和3年6月9日

1 (1) 解答

1.1 (1-1)

$\underbrace{0, 1, 0, 2, 0, 3, 0, 4}_{256}$

1.2 (1-2)

```
main(){
    static int data[1024];
    register int i, j;
    for (j = 0; j < 4; j++)
        for (i = 0; i < 256; i++)
            data[i + j*256] = 0;
}
```

このプログラムの時の参照ストリングは以下になる。

$\underbrace{0, 1}_{256}, \underbrace{0, 2}_{256}, \underbrace{0, 3}_{256}, \underbrace{0, 4}_{256}$

2 (2) 解答

2.1 (2-1)

- 外部フラグメンテーションが発生しやすいという欠点。
- ユーザーが、ファイル生成時に、ファイルサイズを指定することが必須であるという欠点

2.2 (2-2)

図1に示すように、ポインタのリンク情報を別に索引テーブルとしてメモリ上に置くことで、ファイル内容と次ブロックへのポインタとを分離し、次ブロックへのポインタの方をFATというリストとして一括保持する。これにより、リンクファイル割付方式で、リンクをたどるのに各ブロックに順次アクセスする必要があるという欠点が解決されており、直接アクセスする場合に、ブロックアドレスを知るためにディスクにアクセスする必要がなくなり、高速化されている。

2.3 (2-3)

- 利点
ファイル転送の効率が良くなる。
- 欠点
内部フラグメンテーションの度合いが大きくなる。

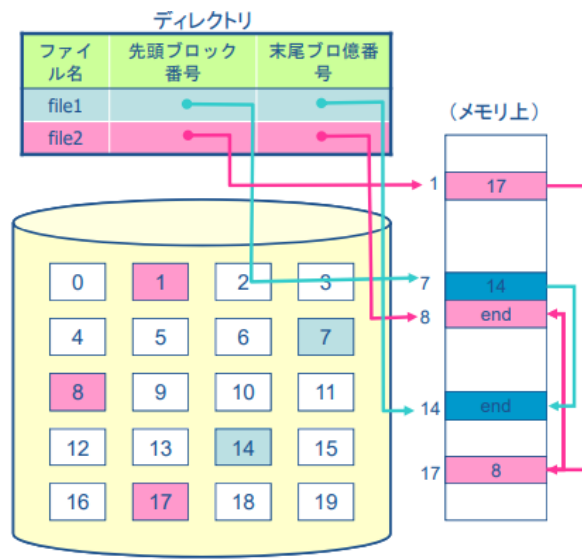


図1 FAT 実装例

2.4 (2-4)

ファイルサイズが大きい場合にアドレスポインタを階層的に管理できるから。

2.5 (2-5)

- 6KB の場合
直接ブロック 6 個 = 6KB
- 220KB の場合
間接ブロック 1 つでさせるデータブロックは 256 個 (ポインタ $4B \times 256$ 個 = 1KB) であり、220 個のブロックが必要なので、直接ブロック 10 個と間接ブロック 1 個でおさめることができる。
よって、合計 221KB
- 500KB の場合
間接ブロック 1 つでさせるデータブロックは 256 個なので、残りを二重間接で保存すれば、500KB をおさめることができる。間接ブロックは、単一間接で 1 つ、二重間接で 2 つ必要なので、合計 503KB