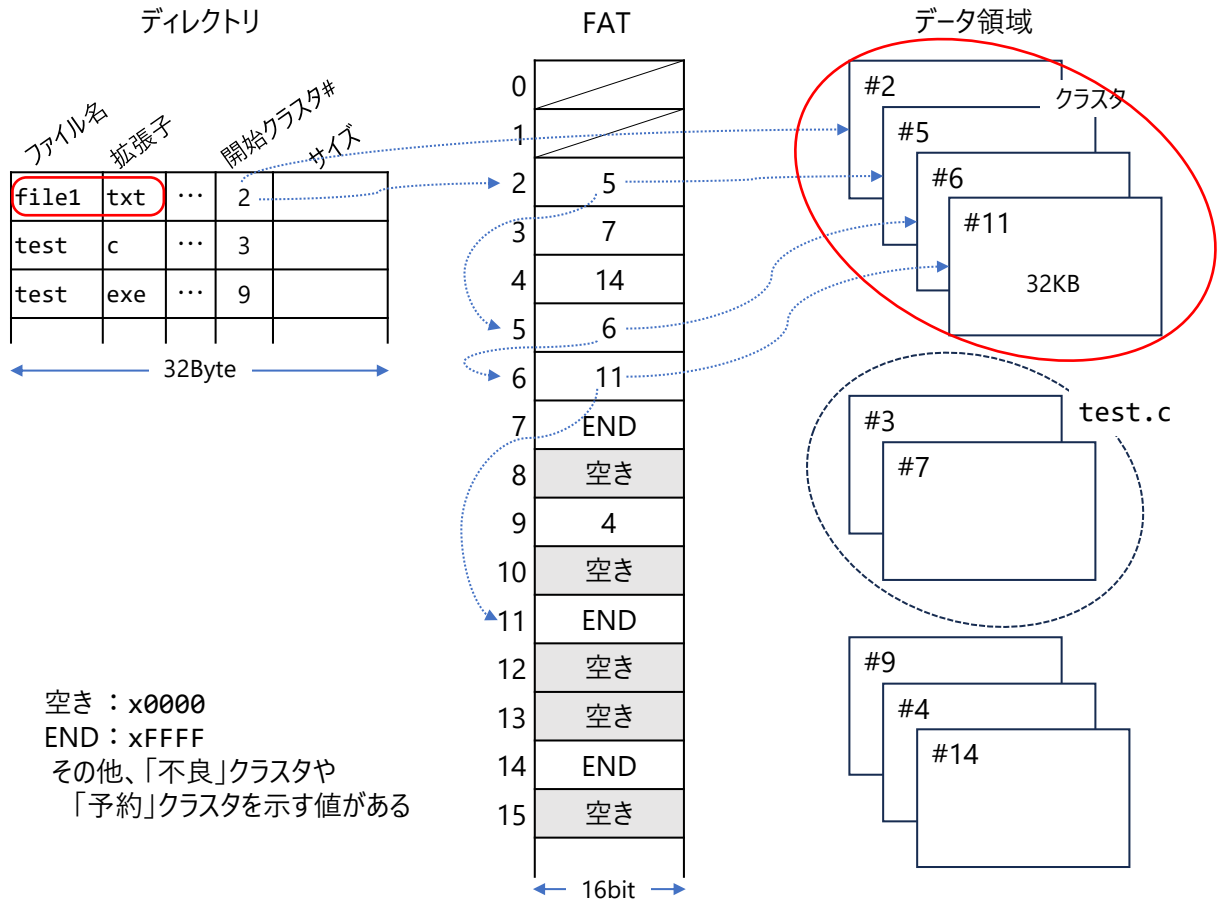
第 8 回の課題について

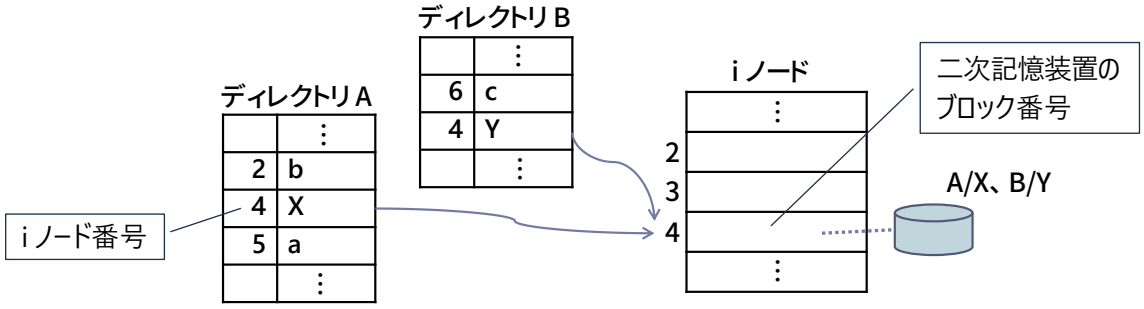
（1）p.19 のファイルの構造で、クラスタ番号が 16 ビットで表現され、1 クラスタサイズが 32KB の場合、1 つのファイルの最大サイズはどうなると考えられるか

FAT のエントリで管理されるクラスタ番号が 16 ビットのサイズであるので、64K（2^{16}=65536）個のクラスタを指定することができ (ただし、「空き」や「END」などを表す値は使えないので、65536 より少ない)、クラスタのサイズが 32KB であるので、64K 個×32KB=2048MB（2GB）となります。

参考までに、p.19 の図の補足として、FAT16 ファイルシステムでの管理の概略を下図に示します。たとえば、名前 file1.txt のファイルが（#2, #5, #6, #11）のクラスタ群で構成されています。



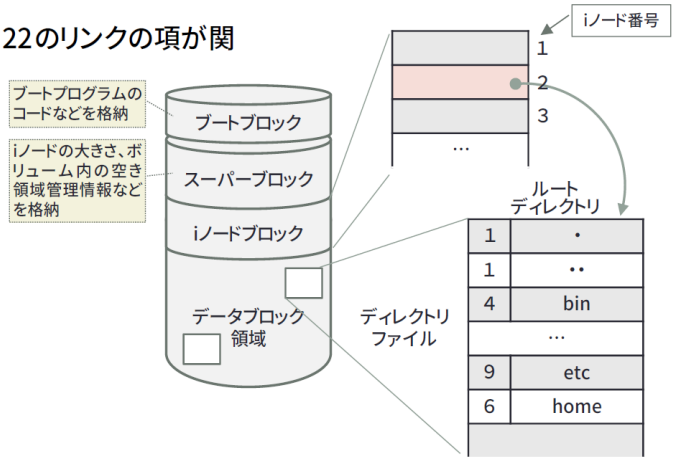
ここのディレクトリと第8回講義資料 p.22 や p.28 にある UNIX のディレクトリとの違いも見ていただくとよいと思います。（以下は、p.22 のもの）



FAT16 では、ボリュームの固定アドレスから固定サイズ（32Byte）エントリでルートディレクトリが開始されますので、ここを探索して対応する開始クラスタ番号が得られます。

UNIX の場合、i ノード領域の固定番号にルートディレクトリが置かれます（第8回資料 p.28）。

- ディレクトリの構造（教科書 p.82）については p.22のリンクの項が関連します
  UNIXの場合、2番目の i ノードがルートディレクトリを指していて、ディレクトリは i ノード番号とファイル名で構成されます



以下は、参考書「オペレーティングシステムの基礎」から引用する UNIX におけるファイル探索の手順です。

> 具体的な事例として，/home/yosi/sdl のファイルを探索する．図 3･25 にその手順を示した．最初にルートディレクトリ「/」内をサーチする．ルートディレクトリは図 3･19 に示したように第 2 番目の i-node でありその実体を読み込むことができる．図 3･25 の左は読み込んだルートディレクトリを示す．ルートディレクトリの中からディレクトリ名「home」を探す．ディレクトリの形式は図 3･20 に示した通りである．

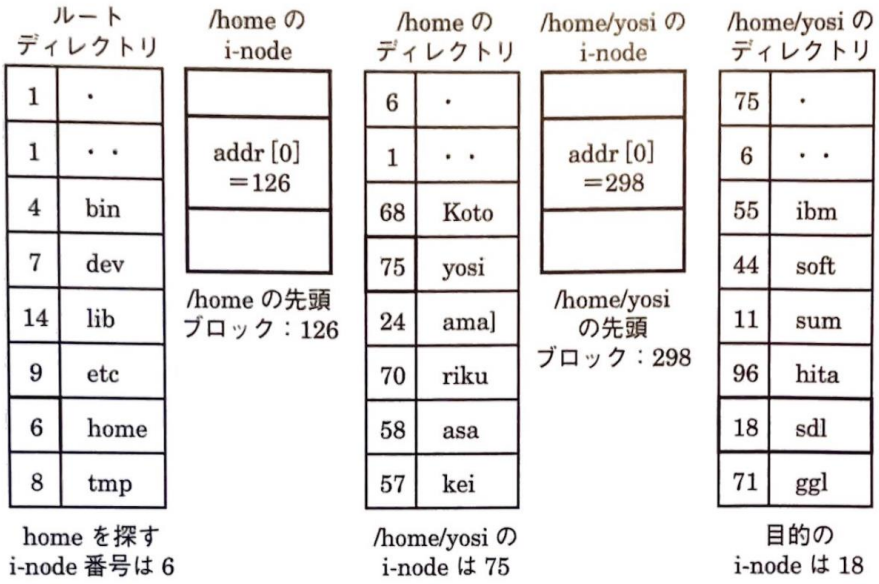


図 3・25　ディレクトリと i-node によるファイル探索の具体例

「home」のエントリから i-node 番号「6」を得る．この結果図 3･21 に示したように,「/home」のディレクトリファイルのブロックアドレスを求めることができる．この例では，ブロックアドレスが「126」である．そこで 126 のブロックを読み込み「/home」のディレクトリ内を探索できるようになる．

/home のディレクトリから次の目標であるファイル名「yosi」を探す．エントリが見つかりその i-node 番号「75」を得る．同様の手順で/home/yosi/の i-node を参照しブロックアドレス「298」を得る．この結果ディレクトリ/home/yosi/の中から目的の「sdl」を探しその i-node 番号「18」を得ることができ/home/yosi/sdl のファイルをアクセスすることが可能となる．

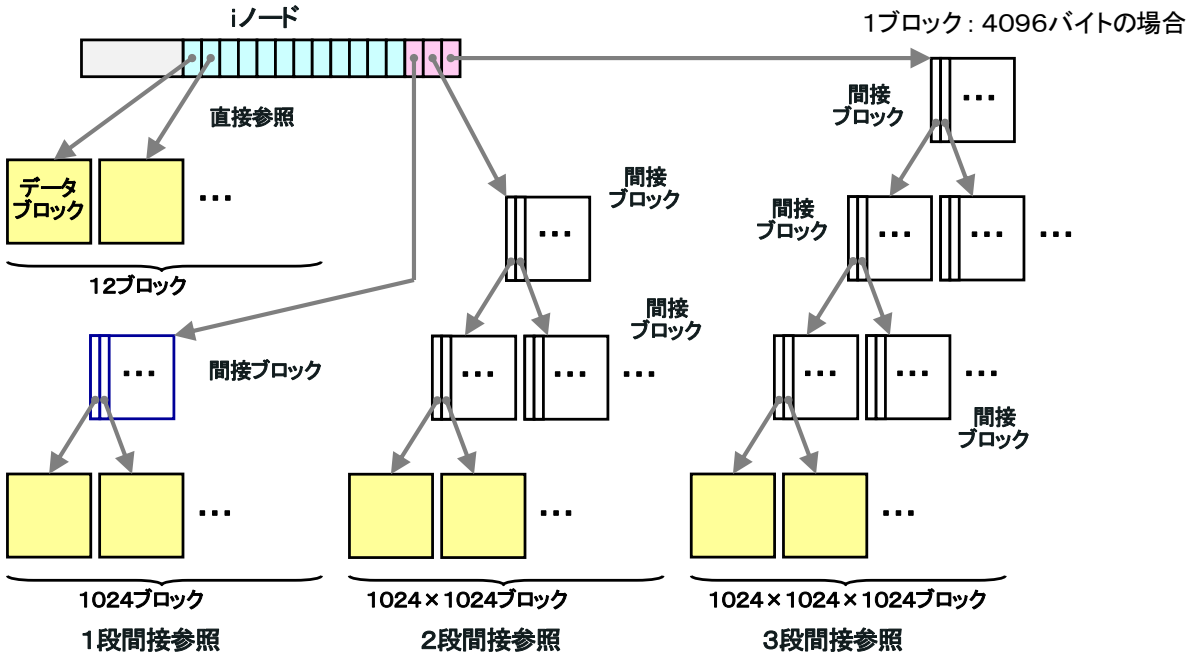
上記の手順により UNIX ファイルは i-node とディレクトリを交互にたどり，目的のファイル実体に到達できる．

（2）p.21 のファイルの構造では、1つのファイルの最大サイズはどうなると考えられるか

データブロックのサイズが 4KB で、p.21 の図で示されているように直接ブロックを参照するエントリは 12 個あるので、直接参照の部分で構成できるファイルのサイズは 12×4KB、一段間接参照では、間接ブロック（サイズは 4KB）がポイントできるブロックは 1024 個（ブロックをポイントするエントリが 4 バイトであるとする）あるので、ここで構成できるファイルのサイズは 1024×4KB＝4MB、二段間接参照の場合は、1 つの間接ブロックにさらに、それぞれもう 1 つ間接ブロックを設けるので、ここで構成できるファイルのサイズは 1024×1024×4KB＝4GB、三段間接参照では、さらにもう 1 段 1024 個の間接ブロックがあるので、ここで構成できるファイルのサイズは 1024×1024×1024×4KB＝4TB となり、最大のサイズはこれらを加算したものになります。（ほとんど三段間接参照による数値ですが、4 つを加算したものです）

$$12\times4\mathrm{KB} + 1024\times4\mathrm{KB} + 1024\times1024\times4\mathrm{KB} + 1024\times1024\times1024\times4\mathrm{KB} \fallingdotseq 4\mathrm{TB}$$

およそ 4TB ということになります。



多くの方に解答いただいたように、講義や教科書の説明の範囲では、上記のような計算ができます。（教科書では間接ブロックのサイズを 1KB としていて 4 バイトのエントリは 256 個となっていますので、値が違ってくることに注意してください）

実際の最大ファイルサイズは、ファイルシステムやハードウェアの制約など環境によって異なることもあるといったような説明をいただいた方もいらっしゃいます。

それに関連しまして、この課題の条件にはありませんが、実際の ext2 や ext3 ファイルシステムでは、ファイルに対して割り当てられているセクタ数（512 バイト単位）を管理しているデータがあります。このデータが 32 ビット長なので、それに制約されてファイルに与えられる最大のセクタ数は 2^{32}-1（約4G 個）となり、ファイルの最大サイズは、4G×0.5KB ＝ 2TB となります。（ext2 や ext3 の諸元に記載されている数字はこれによるものです）

なお、このように大きなファイルに対して間接参照を繰り返して領域を確保するような構造では、

- ファイルサイズが大きくなると、二段や三段の間接参照で数回データブロックにアクセスしなければならないためパフォーマンスが低下する
- ファイルのサイズによって間接ブロックによる参照回数が異なればアクセス時間も変動するので、ファイルを一様に扱いたい場合などでは不都合が生じる

といった問題点があると考えられます。

なお、第 6 回のノート資料にも書きましたが、単位の接頭語として

K（キロ）を 2^{10} = 1024、
M（メガ）を 2^{20} = 1048576、
G（ギガ）を 2^{30} = 1073741824

を表すものなどとしています。