

1) ファイルの名前付けについて、次のことを説明してみよう

ア) ハードディスク(や USB メモリや他の「二次記憶」すべて)において、装置の裸(ソフトなし)の状態を考えると、先頭の0番地から始まるバイトの列である。読み出し・書き込みは、何バイト目ということ指定して行う。(実際は、決まった長さのブロックに分かれていて、ブロック単位で読み・書きする。) このような、何バイト目(=バイト単位で付けたアドレス、「バイトアドレス」と呼ぶ)で指定する読み・書きを使うとすると、実際の操作はようになるだろうか？

(例)

ファイルの名前付けがないシステムだと: ユーザは「欲しい文書データは、このディスクの先頭から 5678123 バイト目から始まる」ということを覚えていて、使い始めるときにアドレス 5678123 を指定して open する。

ファイルの名前付けがあるシステムだと: ユーザは「文書データは report.docx という名前である」ということを覚えていて、使い始める時に、ファイル名 report.docx を指定して open する。

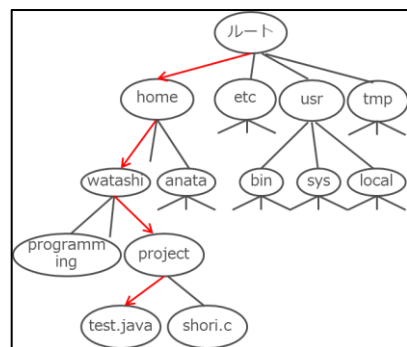
イ) 名前付けでは、すべてのファイルを同じレベルに並べる方法と、フォルダ(ディレクトリ)を用いて階層的に並べる方法がある(右図参照)。これらのメリット・デメリットを比較してみよう。

情報数理 II 演習レポート 1. docx プロジェクト (B) 結果 1. xlsx プロジェクト (B) 結果 2-1. xlsx プロジェクト (B) 結果 2-2. xlsx プログラミング C 課題 1. java プログラミング C 課題 2. java プログラミング C 課題 3. java	情報数理 II 演習レポート 1. docx プロジェクト (B) 実験 1 結果 1. xlsx 実験 2 結果 2-1. xlsx プログラミング C 課題 1. java 課題 2. java 課題 3. java
---	--

ウ) フォルダ(ディレクトリ)を用いて階層的に並べると、形が木(トリー)状になる。この木の形のディレクトリ構造の中で、ほし

いファイルに行きつくために、「パス」を指定する。右図のような木の形の中で、

- ① ルートから最下段の test.java に行きつくためのパスを記述してみよう
- ② 現在見ているフォルダ(ディレクトリ)が右側の local であるとしよう。この時に最下段の test.java に行きつくためのパスを記述してみよう



エ) (時間があれば) 名前のハッシュについて説明してみよう

- ① ハッシュの動作を、文字列のすべての文字の文字コードを足してハッシュ値とする方式を例にして、説明してみよう。
- ② ハッシュを使うと、どういう効果(メリット)があるのか

2) (討論課題) ファイル用の(ハードディスク上の)記憶領域を、ファイルに分割して使う時に起こる問題を議論してみよう

ア) (任意の大きさ=可変長=を割り付けると)、メインメモリの領域管理(複数のプロセスに領域を与えるための管理)と同じように、確保と解放を繰り返すと(外部)フラグメンテーションが起きる。起きる手順を説明してみよう。

イ) (外部)フラグメンテーションを防ぐ方法として、①決まった大きさのブロックに分割し、②そのブロックの番号(アドレス)をアドレス変換することで、不連続のものを連続に見えるようにする、という2つの組合せがある。(メインメモリでも同じ考えだった。) FAT ファイルシステムで、①と②をどのように実現しているか、説明してみよう。

ウ) FAT ファイルシステムは、システムダウンすると、1つのファイルが失われたり、極端にはファイルシステム全体が失われたりした。その理由を考えてみよう。

エ) イ)と同様に、UFS ファイルシステム (i-ノードを使ったファイルシステム)で、①と②をどのように実現しているか、説明してみよう。

オ) UFS ファイルシステムで、直接・1重間接・2重間接・3重間接を混在させるメリットを説明してみよう。