

コンピュータ概論B

－ ソフトウェアを中心に －

#05 ファイル管理とフォーマット

京都産業大学
安田豊

フォーマット（書式）

- ファイルとは
 - － 処理に必要なデータのうち、
 - － メモリに置いておけないものを、
 - － どこかに記録したもの。
- 特徴（メモリの弱点の裏返し）
 - － 電気を切っても保存されている
 - － 大量に記録できる
- 難点（メモリの長所に同じ）
 - － 低速処理
 - － 間接処理（直接CPUが扱えない）

フォーマット（書式）

- フォーマット（教科書 pp.84）
 - － ソフトウェアがある一定の規則にのっとって記録したもの
- 例：アプリケーションソフトウェアのデータファイル
 - － Word の文書ファイル
 - － Excel のワークシート
 - － 画像ファイル
 - － 動画ファイル

フォーマット（書式）

- 規則と書式
 - － Word で文書を書く作業に必要な情報と、
 - － Excel で表計算するのに必要な情報は、
 - － 全く違う
 - － 「一文字めはAで」「A4セルには100で」
- メモリ中のデータも
 - － Word / Excel では全く異なる
- 残されたファイルも
 - － やはり異なるのは当然
 - － 二つのアプリケーションによる二つの規則
 - － 必要な情報をどのように記録するか = 規則 = 書式

フォーマット（書式）

- 互換性
 - － 表現規則を知らなければ復元できない
 - － あるアプリケーションが作ったファイルを別のアプリケーションで開くことが出来るか？
 - － それにはフォーマットを知らなければ
- 「二つのソフトでデータの互換性がある」
 - － 共通のデータフォーマットを利用している
- 質問
 - － Wordがある種の画像ファイルを開くことが出来るのはなぜ？

テキストファイル

- 単に文字列だけを伝えたい場合
 - － テレックスとか（見たこと無いけど）
 - － 電報とか
- 符号化
 - － モールス信号で電報を届ける
 - － 文字を符号化する
 - － 文字は直接電線に乗らない
 - － 直接電線に乗る方法で、間接的に文字を表現する
 - － Aは「ツー」Bは「トン」Cは「ツー、トン」
 - － 文字の符号化ができた
- 脱線：文字自体が符号化の結果なんだけど

テキストファイル

- 通信と符号化規則（プロトコル）
 - ある規則を設けて、それを両端で共有する
- ファイルと符号化規則（フォーマット）
 - ファイルに記録するときと、読み出すときは同じ規則でないダメ
- テレックスの歴史は長い
 - そのまま世界共通の文字符号化ルールになった
 - そのルールを元にしたテキスト（文字）フォーマットは広い互換性をもつ
 - 慣習的にテキストファイルと呼ぶ

ASCII コード表 (7bits code table)

| | | | | | | | | |
|----------|------|---------|--------|--------|--------|--------|---------|---------|
| = 00 | = 16 | SP = 32 | 0 = 48 | @ = 64 | P = 80 | ` = 96 | p = 112 | |
| = 01 | = 17 | ! | = 33 | 1 = 49 | A = 65 | Q = 81 | a = 97 | q = 113 |
| = 02 | = 18 | " | = 34 | 2 = 50 | B = 66 | R = 82 | b = 98 | r = 114 |
| = 03 | = 19 | # | = 35 | 3 = 51 | C = 67 | S = 83 | c = 99 | s = 115 |
| = 04 | = 20 | \$ | = 36 | 4 = 52 | D = 68 | T = 84 | d = 100 | t = 116 |
| = 05 | = 21 | % | = 37 | 5 = 53 | E = 69 | U = 85 | e = 101 | u = 117 |
| = 06 | = 22 | & | = 38 | 6 = 54 | F = 70 | V = 86 | f = 102 | v = 118 |
| = 07 | = 23 | ' | = 39 | 7 = 55 | G = 71 | W = 87 | g = 103 | w = 119 |
| BS = 08 | = 24 | (| = 40 | 8 = 56 | H = 72 | X = 88 | h = 104 | x = 120 |
| TAB = 09 | = 25 |) | = 41 | 9 = 57 | I = 73 | Y = 89 | i = 105 | y = 121 |
| LF = 10 | = 26 | * | = 42 | : = 58 | J = 74 | Z = 90 | j = 106 | z = 122 |
| = 11 | = 27 | + | = 43 | ; = 59 | K = 75 | [= 91 | k = 107 | { = 123 |
| = 12 | = 28 | , | = 44 | < = 60 | L = 76 | \ = 92 | l = 108 | = 124 |
| CR = 13 | = 29 | - | = 45 | = 61 | M = 77 |] = 93 | m = 109 | } = 125 |
| = 14 | = 30 | . | = 46 | > = 62 | N = 78 | ^ = 94 | n = 110 | ~ = 126 |
| = 15 | = 31 | / | = 47 | ? = 63 | O = 79 | _ = 95 | o = 111 | |

TELEX 時代の名残りに注目

- 英文字の方が後ろ
- 印字制御文字がある

バイナリファイル

- テキストファイル以外のファイル書式
 - 中に何が入っているか分からない人にはただの二進データ (binary data)
- 特に実行形式のことを言う (教科書 pp.85)
 - アプリケーションソフトウェア自体もメモリに残せないでファイルとして記録されている
 - これらの実行されるプログラムのファイルをバイナリファイルと呼ぶ場合がある (文脈次第)
 - 恐らく昔はプログラムと文字がファイルの大半を占めていたのでは？

OS / Application / Data

- コンピュータ内ではなんでもファイルとして記録
 - アプリケーションもファイルとして記録
 - OSもファイルとして記録されている
- OS はアプリケーションを実行する
 - アプリケーション・プログラムのファイルをデータとして読み込んで処理
 - Wordが文書ファイルを読み込むのに相似
- プログラムとデータの関係
 - アプリケーション・プログラムはOSのデータファイル
 - 文書ファイルはアプリケーションのデータファイル

たくさんのファイル

- ファイル管理
 - 多くのファイルを扱いやすくユーザに見せる
 - 整理して記録
 - 名前を付ける、記録日付を残す、など
- 拡張子という慣習
 - abc.txt / abc.doc / abc.xls という名前の意味
 - 拡張子でファイルのフォーマットを明示
 - OSは拡張子を見て処理方法を決める
 - 多くの OS で採用された手法 (Windows も)
 - MacOS は別の方法で書式を明示

たくさんのファイル

- 分類方法の工夫
 - 名前、拡張子くらいでは分類しきれない
- 階層的な整理
 - 一群のファイルをまとめる存在
 - 日常生活でもよく使う手法
- 呼び名はイロイロ
 - ディレクトリ: Unix / Windows (ちょっと前)
 - フォルダ: Mac / Windows (95以降)
- 教科書 pp.85-

ファイルシステム

- ビデオテープ（シーケンシャルメディア）
 - 区切りなし、管理はユーザ任せ
- ディスク（ランダムアクセスメディア）
 - データに始まりと終わり（区切り）あり
 - ブロック化して格納
 - 未使用領域の再利用を簡単にするため
 - 最初にブロックの枠切りを行う必要あり
 - 初期化・フォーマットと呼ばれる作業
 - どのブロックに何があるか、別に記録

ファイルシステム

- 余談：
 - CD ROM にブロックは？
 - CD-R は初期化するか？
 - フラグメンテーションって何？

ドライブ

- ドライブ (教科書 pp.85-)
 - ファイルを格納する記憶装置を「ドライブ」と呼ぶ
 - Windows / Mac はドライブが複数ある場合、その違いを意識して使わせている
 - Unix は意識させないタイプ
 - Unified I/O
- さまざまなドライブ
 - HD / FD / CD-ROM / MO
 - それぞれの機器の工夫に注目

用語のまとめ

- ファイルはドライブに格納される
 - ランダムアクセスドライブの場合、ブロック化して管理（再利用が目的）
 - どのブロックに何があるか、別に記録
 - シーケンシャルメディアでもブロック化する場合があるが、再利用目的ではない
- フォルダを利用し、階層化されて保管
 - 便利さのため
 - 階層化ファイルシステムと呼ぶ

OS の機能

- 教科書 pp.83-
 - プロセス管理（プログラム実行管理）
 - メモリ管理（資源管理）
 - ファイル管理
 - 入出力管理（デバイス管理）
 - 通信管理・運用管理などについてもいつか
- 仮想メモリ管理
 - メモリもフラグメントする