

# コンピュータ概論 A/B

-- マインドマップ --  
(思考の視覚化)

数学科 栗野 俊一  
(TA: 高田 健杜, 中村 建太 [院生 1 年])

2019/05/28 コンピュータ概

# 伝言

---

## 私語は慎むように !!

### □ 担任からの連絡

○ 数学科オリエンテーション(軽井沢)は、楽しかったですか？

▶ 大学の友達は一生涯のものになります。是非、在学中に友達を沢山作りましょう

○ 学生証での出席は済ませましたか？

▶ 入口の脇の出席装置に学生証を翳す

### □ 席は自由です

○ できるだけ前に詰めよう

### □ 色々なお知らせについて

○ 栗野の Web Page に注意する事

<http://edu-gw2.math.cst.nihon-u.ac.jp/~kurino>

### □ VNC Server Address : 10.9.209.159

○ Password : vnc-2019

# 前回(2019/05/21)の内容：ファイルシステム

---

## □ 講義内容

### ○ コンピュータ(ハード)の構成

- ▶ CPU : コンピュータの中心部分(計算や処理を行う)
- ▶ Memory : CPU の動作を行うプログラムやデータを記録する(作業領域)
- ▶ I/O (Input/Output) : コンピュータの外との情報のやり取りをする
- ▶ 外部記憶装置(HD) : プログラムを記憶(保存領域) / コンピュータの個性
- ▶ バス : 上記の装置を結び、情報(データ)のやり取りをする通り道

### ○ ファイル : 外部記憶装置に記憶されている「一まとまり」の情報

- ▶ 「記憶されている情報自身」の他に様々な「メタ情報」を含む
- ▶ ファイル名/拡張子/作成日時/持主/etc...

### ○ 拡張子 : ファイル名の一部で、ファイル名の後ろの「.」の後の部分

- ▶ ファイルの種類に対応した拡張子を付ける「習慣」がある / 守られているとは限らない(cf. ウィルス)
- ▶ 「拡張子はアプリケーションと対応付け」されており、「ファイルを開く事ができる」のは、「拡張子が有る」から

### ○ ファイルシステム : ファイルを管理するためのシステム

- ▶ コンピュータが管理し、人間がファイルを扱い易くするためのサービスを提供
- ▶ フォルダ : 複数のファイルやフォルダを一纏めにして名前を付けたもの

# 本日(2019/05/28)の予定

---

- 本日(2019/05/28)の予定
  - 「思考」の視覚化(マインドマップ)
  - 木構造
  - ファイル名と絶対パス名
- 実習
  - [演習 1] freemind の利用
  - [演習 2] 演習課題の作成

# 本日の課題 (2019/05/28)

---

## □ 前回 (2019/05/21) の課題

○しりとりファイル(20190521-QQQQ.txt)のサイズ(byte 単位)を教えてください

▶ファイルの提出ではなく、「サイズ(1234 byte 等)」を直接答えます

## □ 今回 (2019/05/28) の課題

○次のファイルを提出しなさい

▶表題:「私について」というマインドマップ

▶ファイル名: 20190528-QQQQ.mm (QQQQ は学生番号)

▶詳しくは、配布した sample-20190528.mm の内容を参照

▶内容は、「私について」でなくても良く、「何でも」良い

# マインドマップ

---

## □ 知識とは

### ○ 様々な概念や事実と、それらの間の関係(物事)

- ▶ 色々な「物(Entity)」が互いに「関係付け(Relationship)」されている(事)
- ▶ cf. E-R モデル (Entity-Relationship Model)
- ▶ [注意]「織る」は「糸を縦と横に編み込む事」で「識る」は「事実や概念を関連付けする事」

### ○ 「学習する」とは

- ▶ 自分の「知識」に、新しい「概念」や「関連付け」を追加する事
- ▶ [注意] 全く知らない事は解らない / 知っている事を手掛かりに新しい概念を身に付ける

## □ マインドマップとは

### ○ 興味の対象を中心に関連する概念を放射状に記述し、線分で関連付けた図

### ○ 概念間の関係を図示(視覚化)する事により、物事を整理するための思考ツール

- ▶ 概念間の関係の強さを、平面上の距離に対応付け可能
- ▶ ツリー構造による基本的な階層構造の表現

### ○ 「知識」の「可視化」ツール

# 思考の可視化

---

## □ 思考の可視化の必要性

### ○ 基本はコミュニケーションツール

- ▶ 自分のマインド(考え)を他人に提示したい(可視化)

## □ 自分へのコミュニケーション

### ○ 「三日前の自分は他人」の原則

- ▶ 「人」は「忘れる」生物 (自分を疑う習慣)
- ▶ 自分のために「記録」が必要: 「あの時、私は何を考えていた？」
- ▶ !! 様々なツール: 日記、メール、メモ etc..

### ○ 人間の短期記憶の限界 (マジックナンバー 7)

- ▶ cf. 何故、「暗算」は難しいか? / 虹は何故 7 色? / week / 頭が飽和する..

### ○ 短期記憶の内容を外出し(外部記憶装置に記録)する

- ▶ 短期記憶領域を clear にし再利用する事によって、「思考」を継続する

# 木 (Tree)

---

## □ 構造(グラフ)

- いくつかのノード(点/要素)をアーク(線/関係)で結んだもの

## □ 木(き)/木構造(もくこうぞう)

- 階層を表す構造の一つで、次の様に「再帰的」に定義される

- ▶ 単独のノードは木である(この木の根は、この単独のノード自身である)
- ▶ 新しいノードと複数の木の根を結んだ物は木である(根は新しいノード)
- ▶ 上記の二つの規則で作られた物だけが木である

## ○ cf. 「自然数の定義」(ペアノの公理の一部)

- ▶ 1 は自然数である
- ▶  $n$  が自然数ならば、 $n+1$  も自然数である
- ▶ 上記の二つの規則で作られた物だけが自然数である

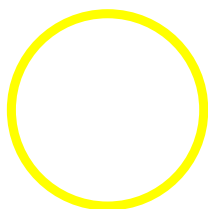
## ○ 木に関連する用語

- ▶ 祖先/子孫：根は他の要素の先祖になる、その逆の関係が子孫
- ▶ 親/子：自分と直接接続している祖先は親、その逆の関係が子
- ▶ 上/下：親が上、子が下 ( 計算機の「木」は根を上を書く )
- ▶ 根/枝/葉：親を持たないノードが根、子を持たないノードが葉、その他が枝

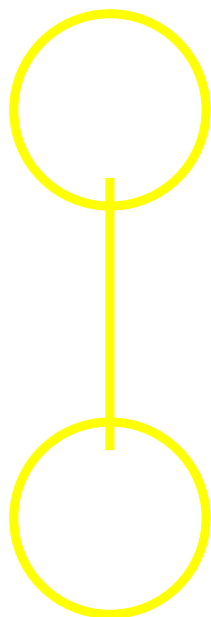


# 木の例

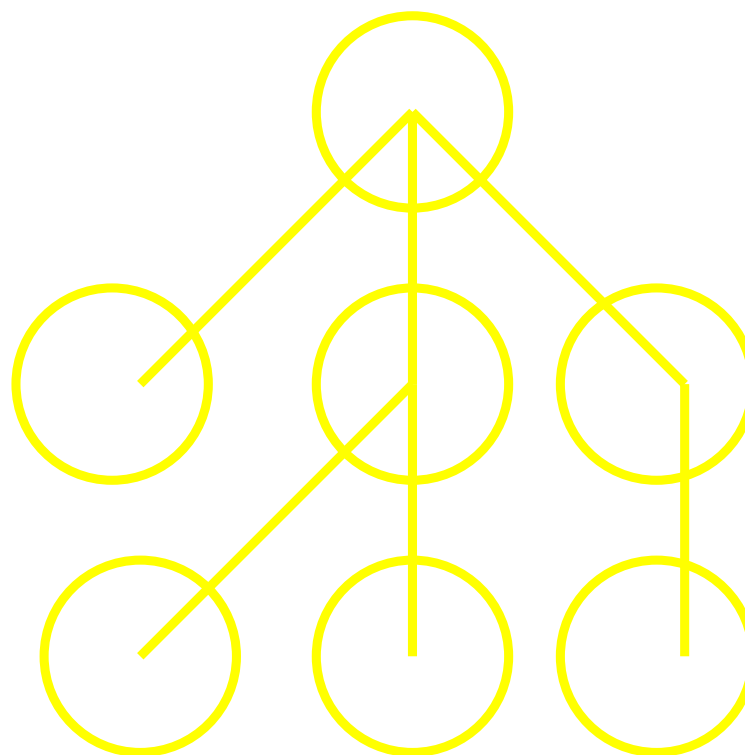
---



(a)



(b)

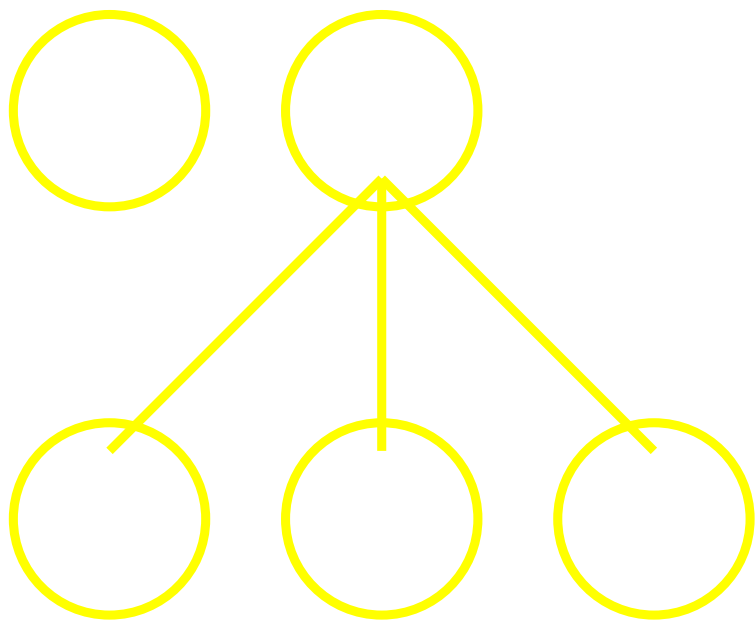


(c)

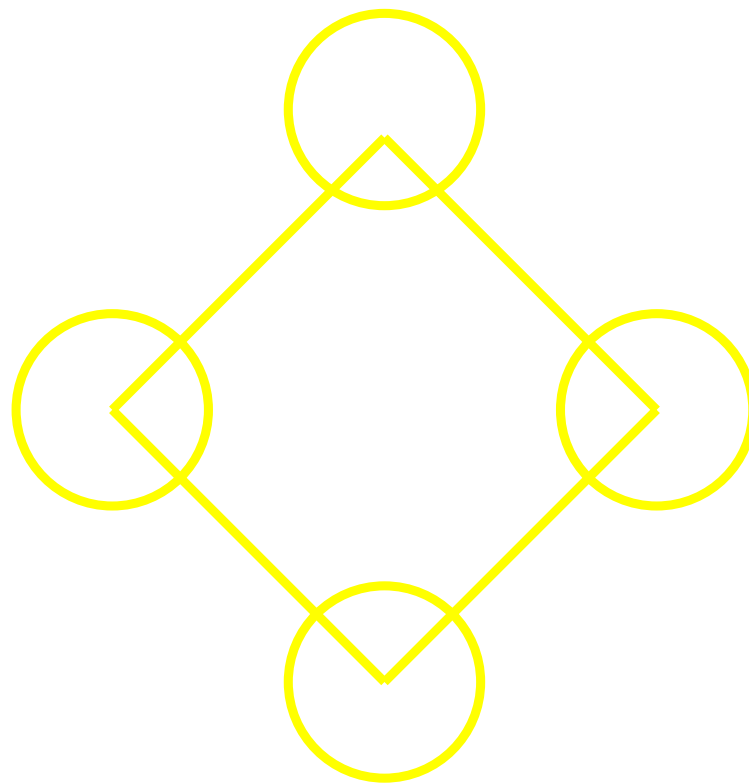
- (a) : 根が一つだけの木
- (b) : 根が一つ、葉が一つで、枝のない木
- (c) : 葉が四つ、枝が三の木

# 木でない例

---



(A)



(B)

- (A) : 繋がってない部分がある(森の例)
- (B) : 輪ができています(木には、輪ができない)

# ファイルシステム

---

## □ ファイルシステム

- 外部記憶上のデータの管理方法の事

  - ▶ ファイルをどのように指定するか的方式

- Windows 10 のファイルシステムは NTFS

- 情報の単位はファイル

## □ NTFS は階層型のファイル管理を行っている

- 外部記憶装置の一番大きな単位は、ドライブ(外部記憶装置)

  - ▶ SSD, HD, DVD-Drive, USB Memory, etc ..

- ドライブ内は木構造

  - ▶ ドライブは一文字 (ドライブ・レター) で表す

  - ▶ ハードディスクは「C」となっている (が、決まっているわけではない)

- 要素(根以外は名前をもつ)は二種類ある

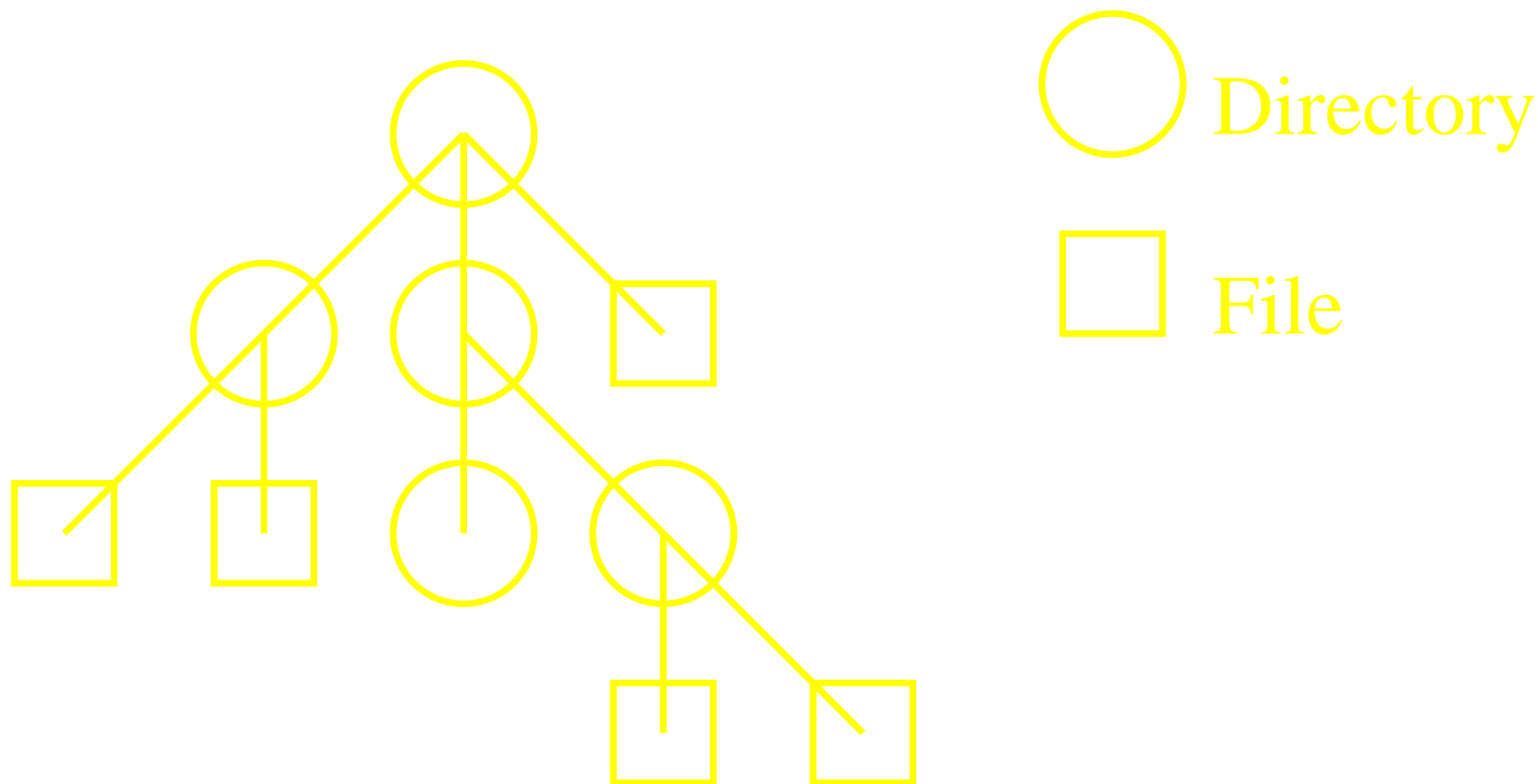
  - ▶ ディレクトリ(フォルダ) : 子を持つ事ができる

  - ▶ ファイル : 子は持てない(必ず葉になる)

- 根の事を「ルート(root)」と呼ぶ事が多い

# ファイルシステムの例

---



- ノードはディレクトリ(○)とファイル(□)の二種類
  - ファイルは葉にしかない (子ができない)
    - ▶ 空のディレクトリも作れ、葉になる
  - 個々のノードには名前がついている

# パス(path)名

---

## □ パス(path)名

- 一つのファイルシステムでファイルを指定するために用いる識別子
  - ▶ そのファイルに「至る道筋(path)」を記述したもの
- パス名の種類
  - ▶ 絶対パス名：共通の出発点(ルート)からの経路を記述
  - ▶ 相対パス名：現在の場所(後日)からの経路を記述

## □ 絶対パス(path)名

- ディレクトリ/ファイルを一意に示す識別子
  - ▶ 根からディレクトリ名を「¥」で継げたもの (かならず「¥」で始まる)
  - ▶ 根からそのノードまでの経路(path)を示す
  - ▶ 根自身のパス名は「¥」
- Windows ファイルシステムのパス名の特徴
  - ▶ 最初はドライブレターから始まる
  - ▶ 「デスクトップ」は、「C:¥Users¥pc¥Desktop」の省略形
- 絶対パス名の例
  - ▶ マインドマップサンプル:「C:¥Users¥pc¥Desktop¥sample-20190528.mm」

# 演習 0 : jdk のインストール/起動確認

---

- [演習 0.1] jdk インストールパッケージの入手
  - Web から入手
- [演習 0.2] jdk のインストール
  - 右クリックから「管理者として実行」
- [演習 0.3] 後は、基本 [Next] ...
- Java
  - 旧 Sun 社 (現 Oracle 社に買収された) で開発された Programming 言語/実行環境の名前
  - FreeMind の実行に Java の環境が必要

# 演習 1 : FreeMind のインストール/起動確認

---

- [演習 1.1] FreeMind インストールパッケージの入手
- [演習 1.2] FreeMind のインストール
  - 右クリックから「管理者として実行」
    - ▶ Java の実行環境 (JRE) もインストールされる事に注意
  - 言語は [English] で良い
  - java のインストールも要求されるので [Yes] を選択
    - ▶ java のインストーラ : [インストール] → [閉じる]
    - ▶ ブラウザが起動したら、一旦、ブラウザを再起動
  - FreeMind のインストーラに戻って [Next] → [accept] → ... → [Finsh]
- [演習 1.3] FreeMind の実行
  - ディスクトップの FreeMind アイコンをクリック

# 演習 2：マインドマップの作成

---

## □ [演習 2.1] マインドマップの作成

### ○ 基本操作

- ▶ [Ins] キー：子供ノードを作る
- ▶ [Enter] キー：兄弟ノードを作る
- ▶ [Delete] キー：ノードの削除
- ▶ [左クリック]/[F2]：ノードの内容を変更する ([Enter] で終了)
- ▶ [右クリック]：メニュー表示
- ▶ [Esc]：現在の作業の中断(命令の取消)

### ○ マップの作成法

- ▶ 中心に基本概念：後は「木(放射)状」に概念を「発展させて」行く
- ▶ 好きな順番で好きな場所に好きな内容を「拡張」して行く
- ▶ 「まとめ」は「後回し」が良い：思い付いた順で構わない

## □ [演習 2.2] 課題提出

- 表題：「私について」というマインドマップ
- ファイル名：20190528-QQQQ.mm (QQQQ は学生番号)
- 詳しくは、配布した sample-20190528.mm の内容を参照
  - ▶ 注意：内容は何でも良い / 「個人情報」は、記載 \*しない\*



# Windows Update Blocker について

---

## □ Windows Update について

- Windows Update は「Windows の更新」で重要
  - ▶ 基本は「やる」事が前提
- しかし、時々、トラブルを起す事がある
  - ▶ 現状の「自動更新」は「危険」

## □ Wub : Windows Update Blocker

- Windows Update の「自動更新」を「ブロック」するソフト
  - ▶ 通常は Block しておき、必要な時に、On にして手動で update する

## □ Wub のインストール

- Wub.zip を展開するだけ
  - ▶ 新しくフォルダに Wub.exe がある

## □ Wub の起動

- Wub.exe を開くだけ

## □ Wub の操作

- Windows Update の 有効/無効を選択するだけ
  - ▶ 通常は「無効」に、時々「有効」にして、「手動」で Windows Update する