



☆情報ノート☆

はじめに

このノートは、主に授業中に先生が強調したところや、ホワイトボードに書いた重要なことをまとめたものです(今日の授業でやったところまでです)。だから、どこが大切だったかを思い出したり、見そびれたホワイトボードの内容を確認したりするのに使ってください。語句の意味については省いているのがほとんどですし、先生が流しただけの部分は触れていないなど、不十分な点も多くあるので、本当にこのノートは試験勉強の導入や、やる気が出ないときに眺めるくらいの目的で使ってください。むしろ、見なくても大丈夫なくらいです(笑) やっぱり一番大切なのは教科書だと思います。あとは、先生はスライドを中心に授業を進めたので、スライドも量がありますが、補足として見ておくといいと思います。シケプリ・単語集も、上手くまとまっていると思うので、活用してみてください。結構範囲は広く、大変ですけど、がんばりましょうねっ☆彡

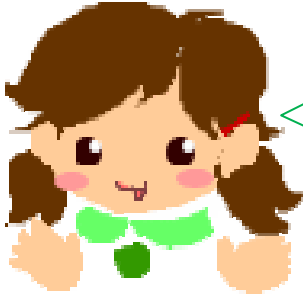
最後に、私の趣味でかわいくしすぎて本当にすみませんでした…。

07S125 H.O

1



第2章 情報の表現



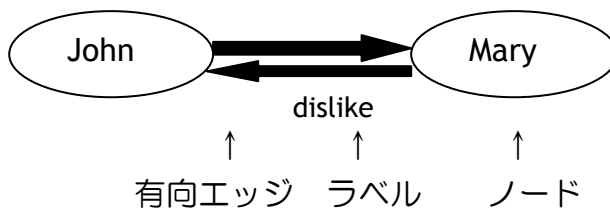
- 表現のいろいろな側面
- ★自然言語と人工言語
 - ★手続き的表現と宣言的表現
 - ★記号表現とパターン表現
 - ★デジタルとアナログ
- 重要語句です◎

言語の3つの側面(構文論・意味論・語用論—スライド参照)も CHECK です！

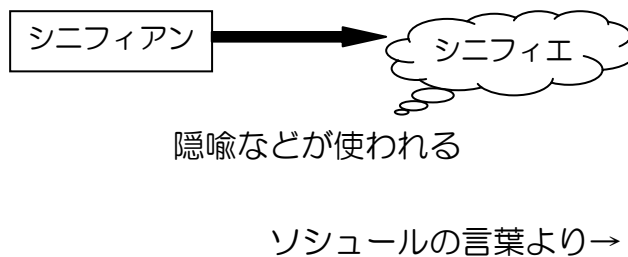
ここからは情報の表現をいろいろと見ていきます。

○モデル・モデル化

- ・モデル化で用いられる表現形式の例 —”ラベル付きグラフ”



○記号表現—記号のもつ2つの側面



ある言葉が指すものは、世界にある実物ではない。その言葉が世界から勝手に切り取ったものである。(分節)言葉が何を指すかは社会的、文化的に決まっているだけである。自然自身の中にそれを必然とする根拠があるわけではない。(言葉の恣意性)

○数の表現

- ・コンピューターで表現できる数値はビット数による(8ビットなら0~255まで)。
- ・2進数⇔10進数はできるようにする。
- ・Gray コード(スライド参照)は前後の数と1ケタしか違わないため、2進数よりも早く計算できる。また、ビットの変化が少ないことを生かして、これは遺伝アルゴリズムにも使われる。

○アナログとデジタル—量子化・標本化が重要！

標本化においては**標本化定理**
(by シャノン)が重要！**ナイキ**
スト周波数や、**エイリアシン**
グ、**モアレ**なども CHECK!!



この章であと絶対に理解すること

- ★ハミング距離
- ★グレイ符号
- ★可逆・非可逆圧縮
- ★ハフマン符号化
- ★ランレングス圧縮
- ★JPEG 圧縮
- ★パリティ検査
- ★ハミング符号

教科書・スライドなどを参考にしてください！

第3章 情報の伝達と通信



まず情報の伝達では知らない程度をへらすことがポイント☆
その上でこの章の情報の伝達に関するところで重要な点は、

- 情報量の加法性
- 情報量の定義… $-\log_2(\text{確率})$
- 平均情報量… $-P_1 * \log_2(P_1) - P_2 * \log_2(P_2)$
- 平均符号長… $\sum(\text{符号長 } i \times \text{確率 } P)$

○情報の通信…**プロトコル**(通信する際の決めごと)が大切。

クライアント ⇔ **サーバ**

・暗号について…**共通鍵暗号**—合鍵という感じ。暗号化も復号化も同じ鍵。

公開鍵暗号—暗号化の鍵は公開、復号化の鍵は秘密にする。

≪公開鍵暗号の利点—復号化が難しいことを表す例≫

$f(x) = Ax^6 + Bx^5 + Cx^4 + \dots + G$ で、 $x=a$ のときの $f(x)$ の値はすぐに求まるけれど、この6次方程式を解くのは難しい。(ワンウェイファンクション)

情報ネットワークのチェックポイント☆

- ★パケット交換
- ★遅延とスループット (遅延は少なく、スループットは大きいほうがいい)
- ★ブロードキャストとユニキャスト
- ★OSI ★IPアドレス ★DNS もちろんインターネットもCHECKです◎

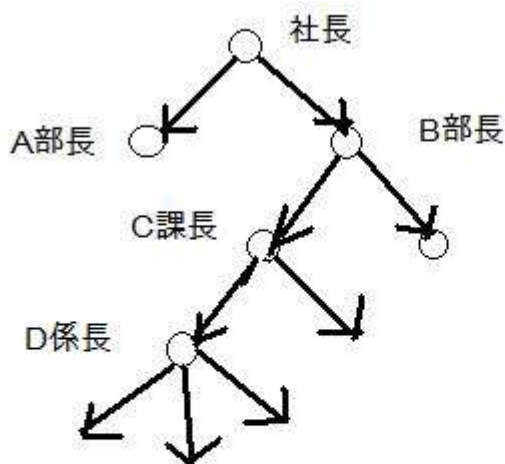
第4章 データの扱い

ここでは、データを体系的に扱うための**データモデル**をいくつか紹介します。

○集合モデル…ベン図

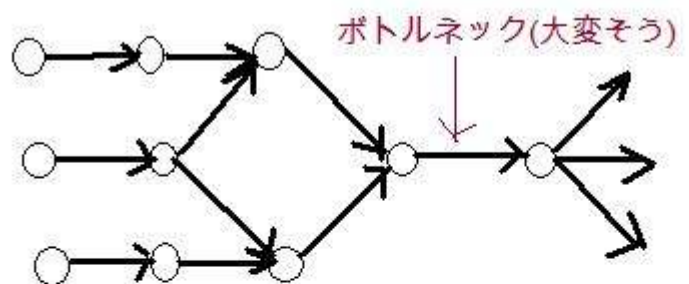
○ネットワークモデル

例：組織図



- ・実体関連モデル(ERモデル)
- ・意味ネットワーク もあり。

PERT図…仕事の流れ、手順を示す。



どの工程が一番大変かがわかる。

○階層モデル…分類を表すのに便利(よくある形)

- ・順序木 ・ゲーム木 ・決定木 など

○関係モデル…世の中を表で表し、これを集めたもの

- ・正規化(#890 など架空のものを使って整理)⇒関係代数が作れる

第5章 計算の方法

☆計算の仕方は主に**取り出し型**と**分割型**の2種類！

☆**再帰**の例

定義

$$n! \equiv 1 \quad \text{if } n \leq 1$$

$$n \times (n-1)! \quad \text{otherwise} \quad \leftarrow ! \text{の定義に!が使われている。}$$

☆計算モデル

・**手続き型** ・**関数型** ・**宣言型** (what you want をかく)

《宣言型の例》

$f(x)=3x^2+5x-1$ をみたす x を求めたいといえれば自動的に求めてくれるモデル

ここで「宣言型」は普通「論理型」と呼ぶ。



プログラム言語(プログラムを記述するための文法のようなもの)における重要単語☆

★機械語 ★バイナリプログラム ★言語処理系

★コンパイラ ★アセンブリ言語 ★アセンブラ

★高水準言語 ★インタプリタ

第6章 問題の解き方

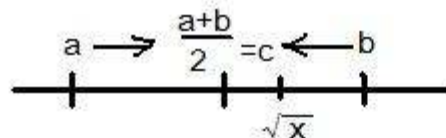
☆**アルゴリズム**…計算手順、問題の解き方



○アルゴリズムの実例

・平方根…**反復法**(時間がかかる)、**二分法**(イメージは下↓)

・フェボナッチ数…**再帰法**(同じ計算を何度もする)、**メモ化法**(同じ計算を繰り返さない)



cf. 再帰関数の例(竹内関数!?)

$f(x,y,z) = \text{if } x > y \text{ then } f(f(x-1,y,z), f(y-1,z,x), f(z-1,x,y))$
else y



計算量—計算時間について☆

★見積もり方

大きさを変数で表す→計算を回数で表す
→オーダーを使う

★空間計算量…どれくらいのメモリを使うか

○計算のモデル

- ・有限状態機械…コンピュータの電子回路を単純化
- ・チューリング機械…無限の記憶装置あり

>(ちょっと劣る)ランダムアクセス機械…有限の記憶装置あり(現実はコレ)

※これらの計算能力は同じ☆

○計算可能性

- ・解ける
- ・解けない
- ・解けるかどうかわからない
- ・モデル化できない ex.チューリングテスト

計算機

プログラム

見て、停止するかどうか判断

ex.Kolatzの問題

n (奇数) $\rightarrow 3n+1$ (偶数) $\rightarrow 2$ で
われるだけわる $\rightarrow n$ (奇数)
を繰り返すと1になる!?

第7章 計算の実現機構

cf.フォンノイマン型コンピュータ…プログラム内臓形式、

プログラムを配線していた時代のコンピュータ



はじめに出てくる重要語句☆

★中央処理装置

…制御装置、演算装置

★主記憶装置(メモリ)

○機械語レベルのプログラム…コンピュータに対する命令の並び

命令コード	オペランド
-------	-------

教科書図 7.2 1001~1011 …プログラム
2001~2003 …データを表す

この図では8種類の命令が
使われています！意味を理
解しておくといいです◎

☆重要ポイント☆

*中央処理装置は論理回路でできている！！



まだまだあるよ☆重要語句☆

★組み合わせ回路 ★順序回路…記憶をもつ

★真理値表

★論理演算 AND(真が2つのときのみ真)

OR(どちらか1つでも真なら真)

NOT(x) (xを否定)

ここからは論理関数について見ていきます！

○論理関数

・完備性…AND、OR、NOTがあればどんな論理関数もかける！

・ANDなどよりもつくりやすい(!?)もの

$NAND(x,y)=NOT(AND(x,y))$ 、 $NOR(x,y)=NOT(OR(x,y))$

両方異なるとき1 ↓ ↓両方が等しいとき1

x	y	XOR(x,y)	EQ(x,y)
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1

EQ=NOT(XOR(x,y))とも表せる。
XORは便利だがANDなどで表すと
 $XOR(x,y)=OR(AND(x,NOT(y)),AND(NOT(x),y))$
となり、ややこしい！

○ブール関数…四則演算のように表す
→標準形に落ち着く

演算法則 $X_1+X_2 \cdot X_3$
 $= (X_1+X_2) \cdot (X_1+X_3)$ に注意！
ド・モルガンの法則等も要CHECK!!

○MIL記法…覚えておく

○演算回路

- ・1ビット半加算器
- ・1ビット全加算器
- ・nビット加算器…全加算器を並べたもの

前の回路の結果を待たないと次の計算ができない→遅れが出る

来週は回路を組み立てるのかな!? 私にはムリです…(苦笑)
みんな来てねー☆彡