

2023.10.27 @福岡県高等学校情報科研究部会

情報科学における 高大接続についての考察

～大学低年次情報教育の経験から～

稲永 俊介

九州大学
システム情報科学研究所
情報学部門

自己紹介

氏名：稲永 俊介 (いねなが しゅんすけ)



1978	福岡県生まれ (現 45歳)
1996	福岡県立 福岡高等学校 卒
1996	九工大 情報工学部 知能情報工学科 卒 (石坂裕毅 研)
2000	九州大学 シス情 情報理学専攻 修士修了 (篠原歩 研)
2002	九州大学 シス情 情報理学専攻 博士学位取得 (主査：有川節夫)
2003	JST ポスドク研究員 (篠原歩 研)
2003	フィンランド ヘルシンキ大学 ポスドク (ロータリー奨学生 兼任) (ホスト研究者：Esko Ukkonen)
2004	JST ポスドク研究員 (篠原歩 研)
2005	京都大学 ポスドク研究員 (岩間一雄 研)
2005	学振 特別研究員 PD (竹田正幸 研)
2006	九州大学 SSP学術研究員 (特任准教授)
2011	九州大学 システム情報科学研究所 情報学部門 准教授
2023～	九州大学 システム情報科学研究所 情報学部門 教授



D1で学位取得
(早期修了)

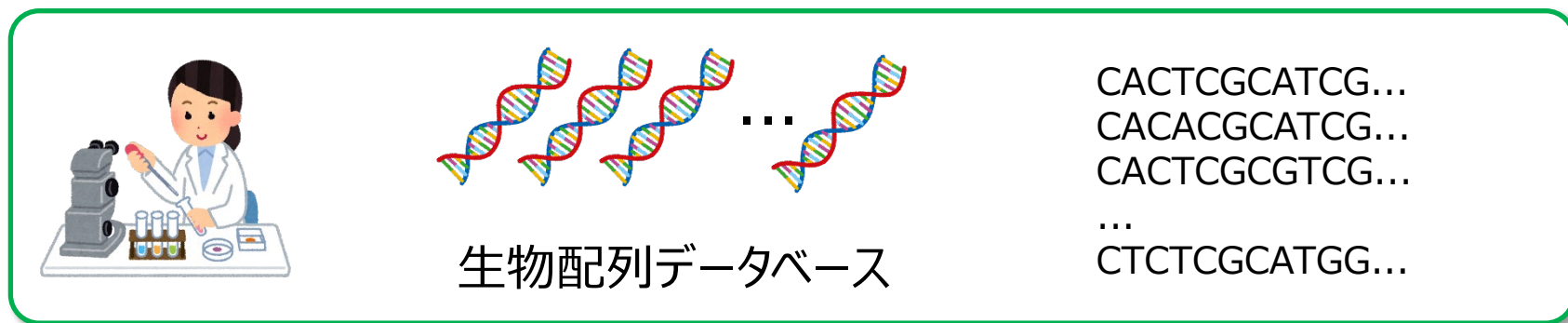
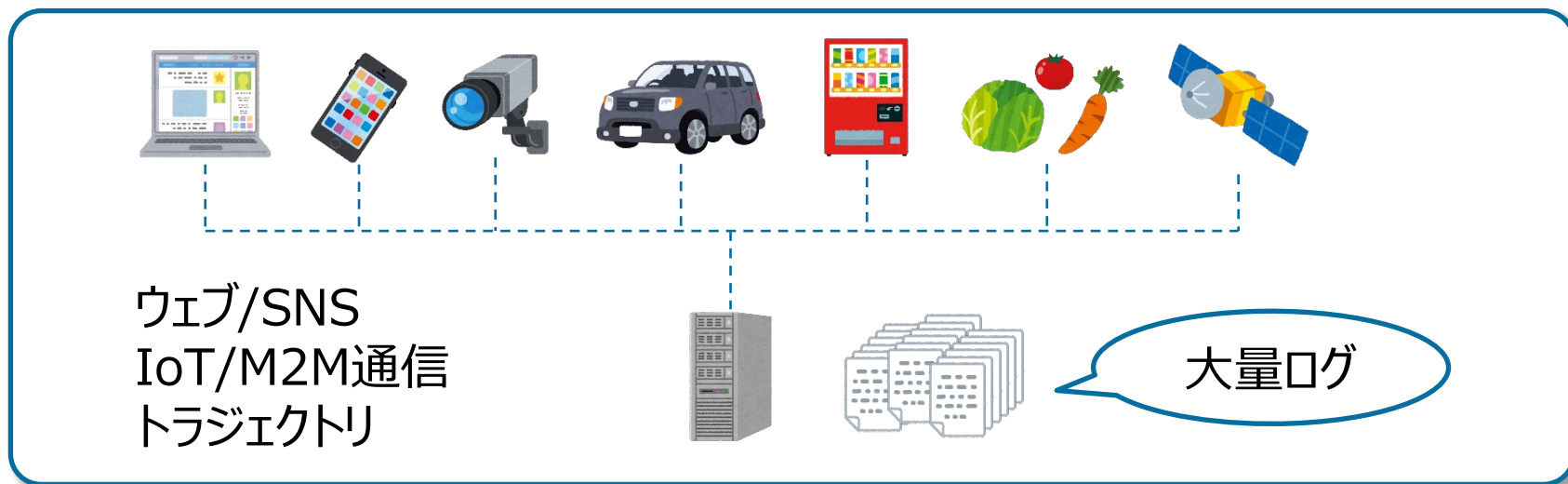
国内・海外
ポスドク
(武者修行時代)

テニユアトラック

准教授

教授

専門分野：文字列処理アルゴリズム，情報検索



多様で膨大な系列データの効率的処理を実現
→ 文字列学 (stringology)

専門分野：文字列処理アルゴリズム，情報検索

情報科学

理論計算機科学

文字列学 (stringology)

処理系

アルゴリズム
データ構造

オートマトン
言語理論

グラフ理論

圧縮系

組合せ論

符号理論

情報理論

証明系

計算可能性

計算量理論

論理数学

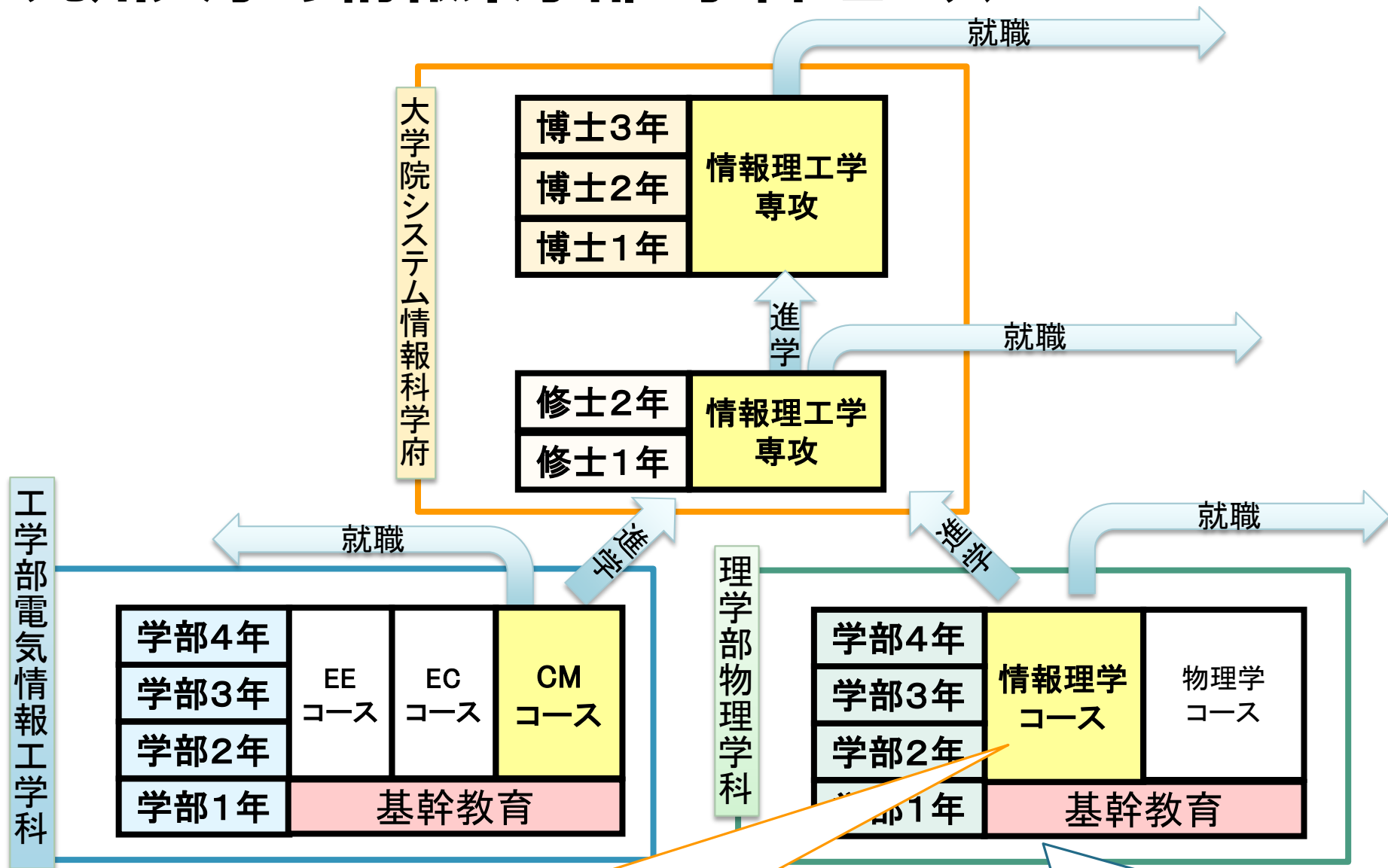
応用系

情報検索

バイオインフォ
マティクス

自然言語処理

九州大学の情報系学部・学科・コース



九州大学 物理学科 情報理学コースの
高校生への周知を是非お願いします!!

- ・ プログラミング演習
- ・ **情報科学**

九大 基幹教育「情報科学」(物理学科1年生)

本日の
講演

	授業のテーマ	授業の内容(90分授業=2時間)	事前/事後学修の内容
1	イントロダクション	本講義全体の導入	講義資料による予習・復習
2	電卓の能力	電卓計算可能な式	講義資料による予習・復習
3	情報の表現	計算機上での情報の表現方法	講義資料による予習・復習
4	偽コイン発見問題	アルゴリズム入門I	講義資料による予習・復習
5	ユークリッドの互除法	アルゴリズム入門II	講義資料による予習・復習
6	ソートと計算時間評価	バブルソート, 選択ソート, オーダー表記	講義資料による予習・復習
7	ヒープソート	ヒープソートアルゴリズムとその時間計算量解析	講義資料による予習・復習
8	マージソート	マージソートアルゴリズムとその時間計算量解析	講義資料による予習・復習
9	比較によらないソート	計数ソート, バケツソート, 基数ソール	講義資料による予習・復習
10	2分探索とその応用	2分探索と文字列パターン照合への応用	講義資料による予習・復習
11	暗号理論入門	シーザー暗号, RSA暗号	講義資料による予習・復習
12	計算の限界	計算モデルなど	講義資料による予習・復習
13	石選び問題	枝刈り法など	講義資料による予習・復習
14	P問題とNP問題	多項式時間で解ける問題, 難しそうな問題のクラス	講義資料による予習・復習
15	全体のまとめ	講義全体のまとめ	講義資料による予習・復習

高校の情報科目とのいちばんの違い

大学の情報系科目では **数学** を駆使する

- 例) 整数, 集合, 論理, 確率, 統計など

計算理論 : アルゴリズムの正しさや計算時間を
論理的に証明 → **プログラム性能保証**

情報理論 : 確率統計に基づく情報エントロピー,
データ圧縮, 暗号理論 → **情報通信技術**

人工知能理論 : 機械学習のための微分積分,
線形代数, 確率統計 → **AI技術**

数学抜きで情報科目を正しく理解することはできない

整数の表現

通常は, 固定長の2進数として表現

例: 8ビット (=1バイト) 長による自然数の表現

整数	内部表現
0	00000000
1	00000001
2	00000010
3	00000011
4	00000100
5	00000101

...

整数	内部表現
250	11111010
251	11111011
252	11111100
253	11111101
254	11111110
255	11111111

0も仲間に
入れて下さい

自然数（符号なし整数）の表現

2進数表記について

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	1	0	1	1	0	0	0	0	1	0	1

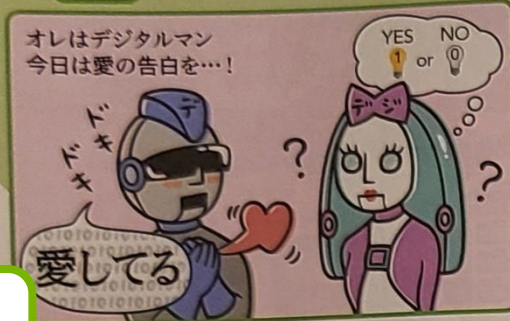
$$2^{10} + 2^8 + 2^7 + 2^2 + 2^0 = 1413_{(10)}$$

- ◆ ビット列 $a_{n-1} a_{n-2} \cdots a_0$ ($a_k \in \{0,1\}$) は
自然数 $\sum_{k=0}^{n-1} a_k 2^k$ を表す

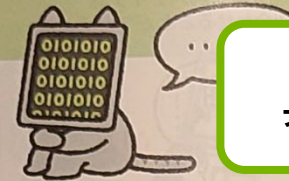
符号なし整数の2進数表記の「数学的な定義」

東京書籍「情報I」

14 数値と文字のデジタル表現



0と1の数字だけでさまざまな数値や文字を表すことができるのはなぜだろうか。



具体例による説明

4ビットの一覧表

2進法の計算 174ページ

表1 2進法と16進法

10進法	2進法	16進法
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F
16	10000	10

1 2進法 → 図1 表1

0から9までの10種類の数字で数値を表現する方法を10進法という。それに対して、コンピュータのように0と1の2つの数字だけで数値を表現する方法を2進法という。10進法で表されたような数も、規則に従えば2進法に変換することができる。2つの数字を組み合わせるだけで、なぜさまざまな情報を表現することができるのか考えてみよう。

2進法1桁では「YES/NO」「表/裏」「上/下」のように2通りまでの情報を表現することができる。2桁と3桁の場合を考えてみよう。

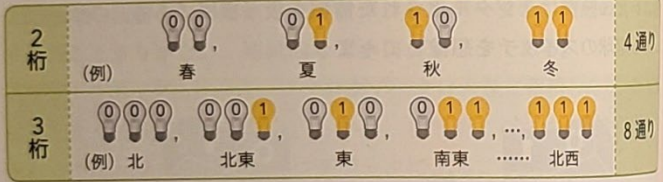


図1 2進法の2桁と3桁の表現

イラスト

コラム 2進法の利点

数研出版「情報I」

具体例による
2進数の説明

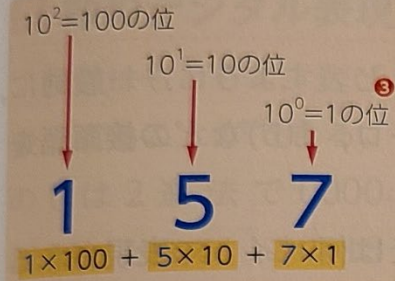


図4 10進法での数の表し方

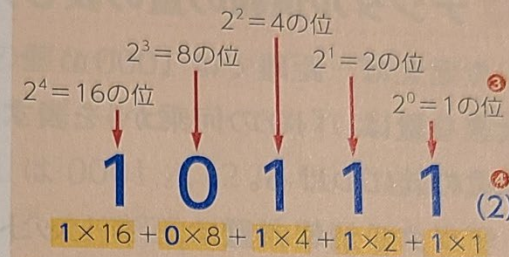


図5 2進法での数の表し方

d 10進法と2進法の変換

2進法で表された数(例: $10111_{(2)}$)を, 10進法に変換するには, 図7のようにする。逆に, 10進法で表された数(例: 23)を2進法に変換するには, 図8のように10進法で表された数を, 順に2でわっていき, 余りを記録して, 下から順に並べる。

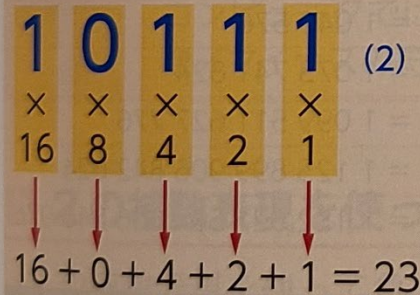


図7 2進法から10進法への変換

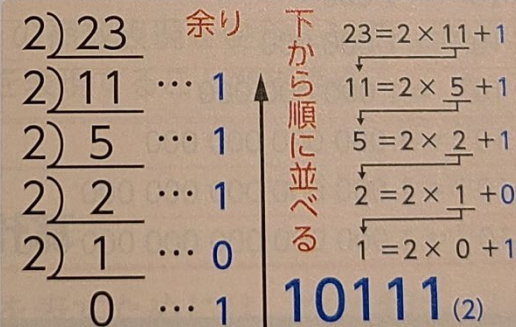


図8 10進法から2進法への変換

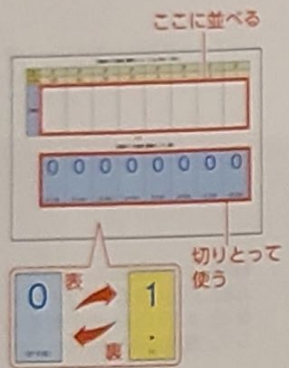
具体例による
2進数 \Leftrightarrow 10進数の
変換法の説明

数研出版「情報I」

道具(カード)を用いて、
変換法を直感的に説明
(巻末に付録あり)

- 必要なもの**
- はさみ
 - 核算用のパソコンなど
 - 2進法⇄10進法変換カード(巻末付録)

Jump ▶
デジタル情報の表現
→ p.50, 51



ヒント

パソコンの「電卓」機能では、「DEC」を選ぶと10進法、「BIN」を選ぶと2進法の表示ができる。

実習 1 10進法の11を2進法に変換しよう。

1 シートの「2進法」の欄にカードを並べよう
「0」を表にして、「位」の欄とカードの位があうようにカードを並べる。

2進法⇄10進法変換シート (p.164~165)

	2^3	2^2	2^1	2^0	2^1	2^0	2^0
10進法	128	64	32	16	8	4	2
2進法	0	0	0	0	0	0	0
	(2^7 の位)	(2^6 の位)	(2^5 の位)	(2^4 の位)	(2^3 の位)	(2^2 の位)	(2^1 の位)

ここをあわせる

2 シート上のカードを裏がえそう
「10進法」の欄を見て、合計が「11」になるようにする。

- 「10進法」の欄を見て、「11」以下の中で最大の「 2^3 」(= 8)のカードを裏がえして、「1」を表にする。
- 「11」から「8」をひいた残りは「3」である。そこで、「3」以下の中で最大の「 2^1 」(= 2)のカードを裏がえして、「1」を表にする。
- 同様に、合計が「11」になるまでカードを裏がえす。

2^3	2^2	2^1	2^0
8	4	2	1

「11」以下の値の中から、最大の「8」をさがす

2進法⇄10進法変換シート (p.164~165)

	2^3	2^2	2^1	2^0	2^1	2^0	2^0
10進法	128	64	32	16	8	4	2
2進法	0	0	0	0	1	0	1
	(2^7 の位)	(2^6 の位)	(2^5 の位)	(2^4 の位)	(2^3 の位)	(2^2 の位)	(2^1 の位)

桁上がりしたら
カードをフリップ

日本文教出版「情報I」

具体例による2進数の説明

具体例による 2進数 \leftrightarrow 10進数の 変換法の説明

TB	テラバイト	1 TB = 1,024 GB (= 2^{40} B)
PB	ペタバイト	1 PB = 1,024 TB (= 2^{50} B)

で用いられる数の表現

0から9までの10種類の数を使う10進法を用いる。いっぽう、コンピュータは2進法の0と1の2つで表現している。たとえば、10進法で「9」という数を表現すると、内部では2進法の「1001」として扱う。10進法であらわされた数は、次のように相互に変換できる。

10進法 → 2進法

2)	19	余り
2)	9	1
2)	4	1
2)	2	0
2)	1	0
	0	1

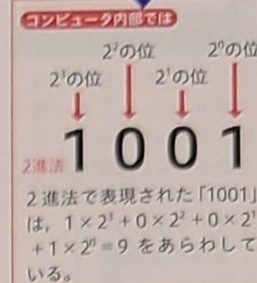
1 0 0 1 1₍₂₎

2進法 → 10進法

1	0	0	1	1				
×	×	×	×	×				
2 ⁴	2 ³	2 ²	2 ¹	2 ⁰				
↓	↓	↓	↓	↓				
16	+	0	+	0	+	2	+	1

19₍₁₀₎

2進法の表現は、10進法に比べて桁数が大きくなる。桁数が大きくなると人間には扱いづらくなるため、プログラミング言語などで記述する際は、このような2進法の数を4ビットずつまとめ、0から9、AからFまでの16種類の記号を用いて表現する(16進法)こともある。



◆2進, 10進, 16進法の対応

10進法	2進法	16進法
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F
16	10000	10

◆n進法の表記方法

ここでは、2進法の10011を10011₍₂₎、10進法の19を19₍₁₀₎とあらわしているが、それぞれ(10011)₂、(19)₁₀とあらわすこともできる。

実教出版「最新情報I」

4ビットの一覧表

10進数	2進数	16進数
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

① 2進数の1001を10進数と区別するために、本書では、 $1001_{(2)}$ のように表記する。このほかに $(1001)_2$ と表記する場合もある。なお、10進数や16進数などについても同様である。

+α

●指数の計算

a の n 乗 (a^n) は、 a を n 回掛け合わせる数を表現している。

$2^0 = 1$ (注) 0ではない
 $2^1 = 2$
 $2^2 = 2 \times 2 = 4$
 $2^3 = 2 \times 2 \times 2 = 8$ となる。

指数関数の定義

とができる。2進数で数値を表現する際には、桁の重みを下位(右側)から4桁ずつに区切って表現する。この方法がよく使われる。16進数では、16進数の0~9の数字とA~Fの英字を使用する。

1桁上がるごとに各桁の重みは、10進数では10倍、2進数では2倍、16進数では16倍になる。

アルゴリズムによる説明

例題 1 2進数と10進数の関係を調べよう

10進数の11は、2進数の1011であることを確かめなさい。

考え方 10進数を2で割り、商が1になるまで割って、商と余りを並べることによって2進数に変換する。

解答

$11_{(10)}$ の2進数への変換

2) 11 余り 1
 2) 5 ... 1 下の桁
 2) 2 ... 1
 1 ... 0 上の桁
 $= 1011_{(2)}$ となる。

$1011_{(2)}$ の10進数への変換

$1011_{(2)} = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$
 $= 1 \times 8 + 0 \times 4 + 1 \times 2 + 1 \times 1$
 $= 8 + 0 + 2 + 1$
 $= 11$
 $11_{(10)}$ となる。

考察 $2^0 = 1$, $10^0 = 1$ であることに注意する。

確認問題

10進数は2進数に、2進数は10進数に変換しなさい。

- (1) $18_{(10)}$ (2) $25_{(10)}$ (3) $32_{(10)}$ (4) $00101011_{(2)}$
 (5) $11100101_{(2)}$ (6) $11111101_{(2)}$

$2^0 = 1$

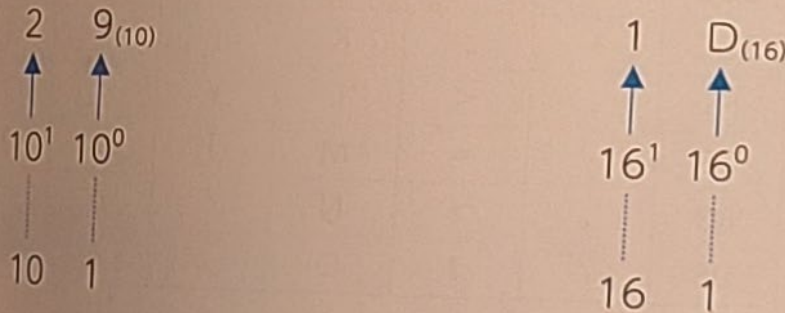
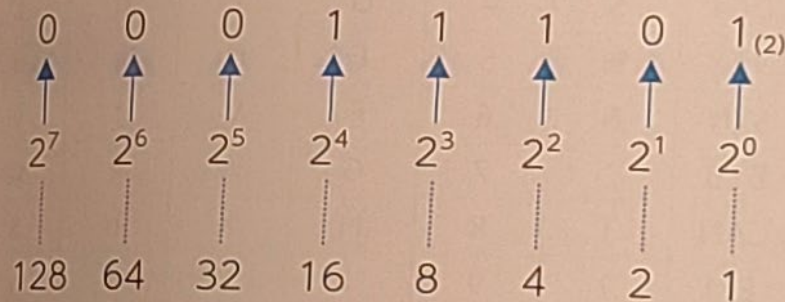
$2^2 = 2 \times 2$
 $2^1 = 2$
 $2^0 = 1$



実教出版「最新情報I」

要点 2進数, 10進数, 16進数の桁の重み

$$00011101_{(2)} \rightarrow 29_{(10)}, 1D_{(16)}$$



自然言語による
 n 進数の定義

n 進数では1桁上がるごとに桁の重みが n 倍になるのか。



確認問題

16進数は2進数に, 2進数は16進数に変換しなさい。
(1) 5F (2) 2F

すべては計算である

コンピュータ上で行われていることはすべて「計算」である。

その「計算」と上手に付き合ったり,
「計算」をうまく制御するには,
「計算」とは何者であるかを理解する必要がある。

そのための科学が

コンピュータ・サイエンス (計算機科学 / 情報科学)
である。

そもそも「計算」とは？

例：プログラミングコンテストの課題

「円周率の小数点以下100桁までを求めよ」

```
print("3.141592653589793238462643383279  
502884197169399375105820974944  
592307816406286208998628034825  
3421170679")
```

は OK ? インチキ ?

「計算」とは、の前に. . .

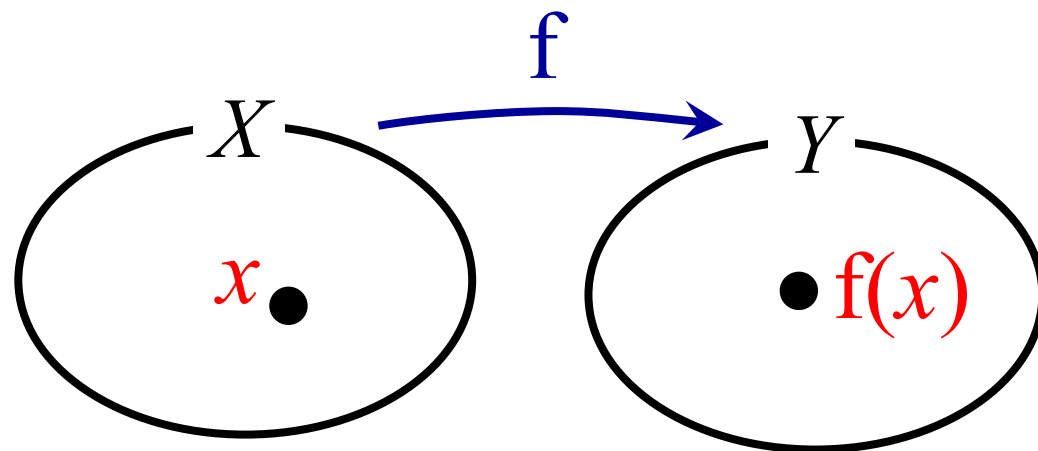
情報科学で扱う「問題」とは？

- 数学的に厳密に定義された関数

問題: $f: X \rightarrow Y$

入力: $x \in X$

出力: $f(x)$



任意の問題事例 $x \in X$ に対し、
答えが一意に定まっていること

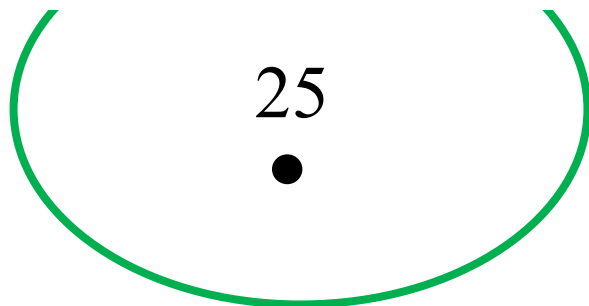
2進数表現を振り返る

問題：2進数から10進数への変換

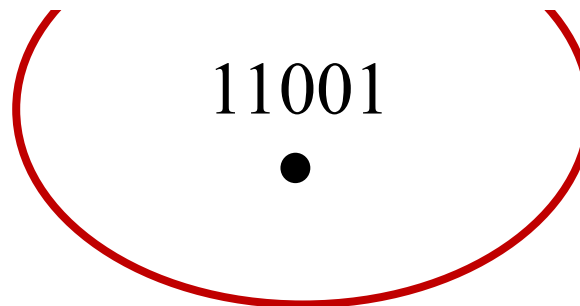
入力：整数 x の10進数表現

出力：整数 x の2進数表現

自然数の
10進数表現の集合 X



自然数の
2進数表現の集合 Y



2進数表現を振り返る

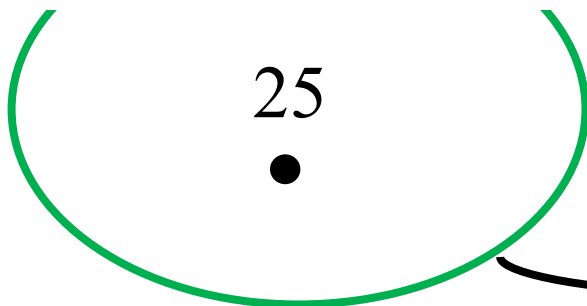
問題：2進数から10進数への変換

入力：整数 x の10進数表現

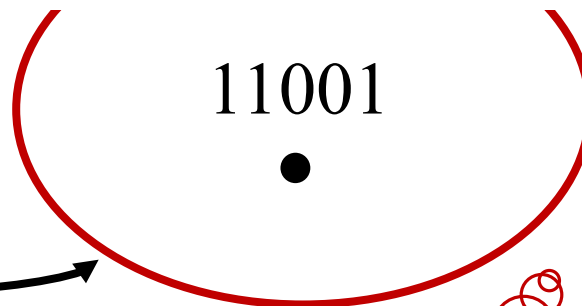
出力：整数 x の2進数表現

2進数表現の定義をせずに
アルゴリズムを説明するのは
数学的におかしい

自然数の
10進数表現の集合 X



自然数の
2進数表現の集合 Y



f

変換アルゴリズムは
関数 f の計算手順のこと

この集合 Y を
定義してからでないと
関数 f は定義できない

2進数表現を振り返る

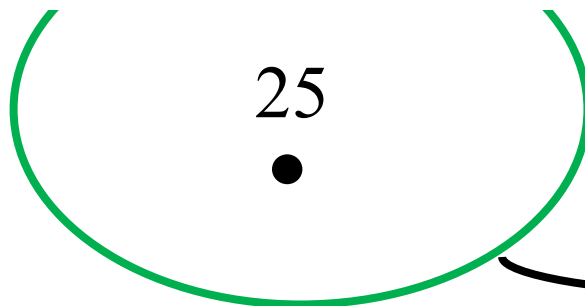
問題：2進数から10進数への変換

入力：整数 x の10進数表現

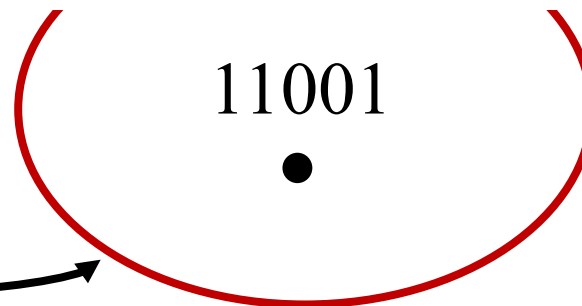
出力：整数 x の2進数表現

2進数表現の定義をせずに
アルゴリズムを説明するのは
数学的におかしい

自然数の
10進数表現の集合 X



自然数の
2進数表現の集合 Y



f

【再掲】

Y の定義

変換アルゴリズムは
関数 f の計算手順のこと

◆ ビット列 $a_{n-1} a_{n-2} \cdots a_0$ ($a_k \in \{0,1\}$) は
自然数 $\sum_{k=0}^{n-1} a_k 2^k$ を表す

例題を振り返る

例：「円周率の小数点以下100桁までを求めよ」

入力：なし

出力：円周率の小数点以下100桁まで

```
print("3.141592653589793238462643383279  
502884197169399375105820974944  
592307816406286208998628034825  
3421170679")
```

は正しく計算している！ ← 定義を満たしている

ここまでのまとめと、ご提案

- コンピュータ上で行われていることはすべて「**計算**」である。
- 「**計算**」とは「**問題**」を解くことである。
- 「**問題**」とは入力の集合から出力の集合への関数である。
- 「**アルゴリズム**」とは「**問題**」を解く手続き、すなわち関数の計算方法である。
- 「**プログラム**」とは「**アルゴリズム**」をコンピュータが解釈可能な言語で記述したものである。

高校の情報科目の授業においても、可能かつ適切な範囲で、数学的な背景があることを、生徒さんに伝えて欲しいです。