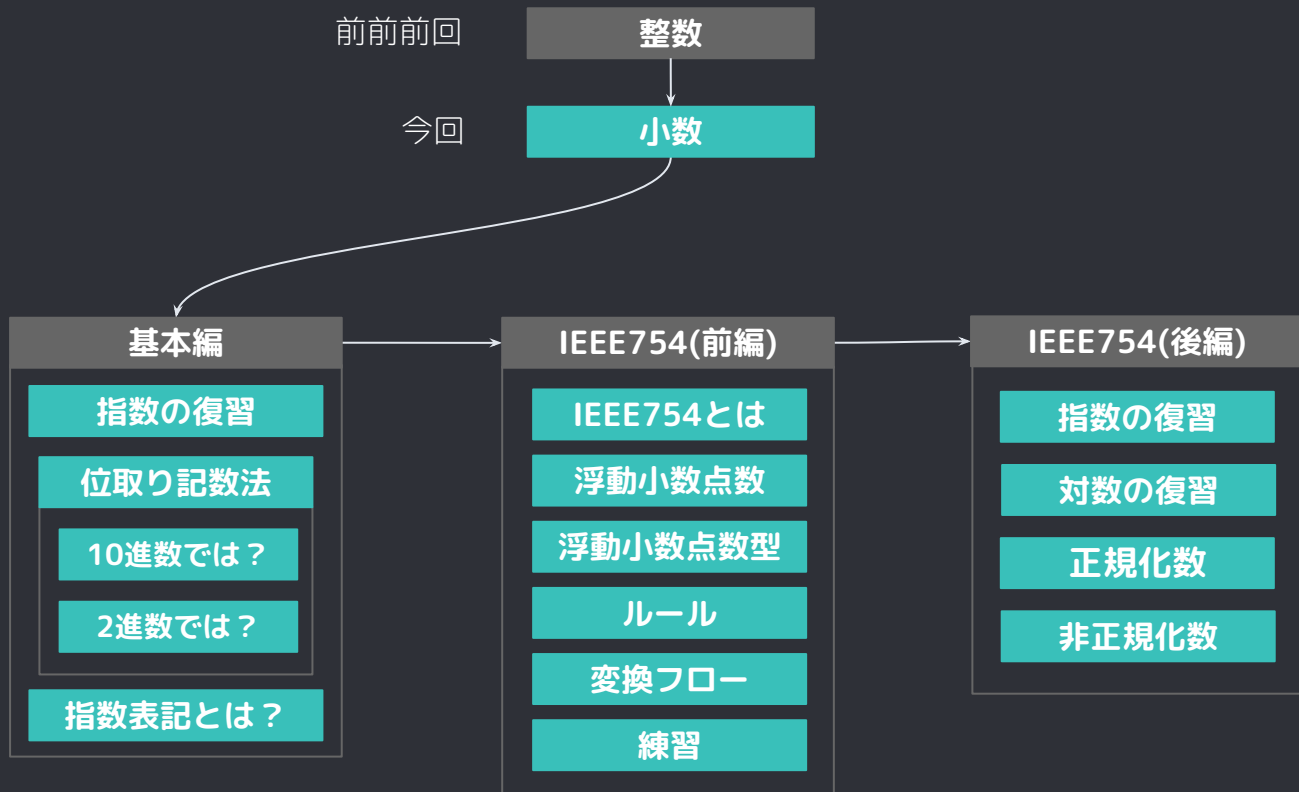




2進数で小数を表す (IEEE754 前編)

プログラマのためのC言語 第10回

概要



“

IEEE754とは？

- IEEE754とは？

- - ✓ 浮動小数点数を扱う上でのルールを定めたもの(標準規格)

● IEEE754とは？

- - ✓ 浮動小数点数を扱う上でのルールを定めたもの(標準規格)
 - ✓ ほぼ全てのモダンなシステムで採用されている

● IEEE754とは？

- ✓ 浮動小数点数を扱う上でのルールを定めたもの(標準規格)
- ✓ ほぼ全てのモダンなシステムで採用されている

昔はカオスじゃった



● IEEE754とは？

- ✓ 浮動小数点数を扱う上でのルールを定めたもの(標準規格)
- ✓ ほぼ全てのモダンなシステムで採用されている
- ✓ IEEE754で定義されていること

● IEEE754とは？

- - ✓ 浮動小数点数を扱う上でのルールを定めたもの(標準規格)
 - ✓ ほぼ全てのモダンなシステムで採用されている
 - ✓ IEEE754で定義されていること
 - ・ 小数をどうやって表すか
 - ・ 0や無限、非数(NaN)の表現方法
 - ・ 端数の丸め規則や、0除算などの例外時の振る舞い
 - ・ 四則演算
 - ・ etc...

● IEEE754とは？

- ✓ 浮動小数点数を扱う上でのルールを定めたもの(標準規格)
- ✓ ほぼ全てのモダンなシステムで採用されている
- ✓ IEEE754で定義されていること
 - ・ 小数をどうやって表すか
 - ・ 0や無限、非数(NaN)の表現方法
 - ・ 端数の丸め規則や、0除算などの例外時の振る舞い
 - ・ 四則演算
 - ・ etc...
- ✓ 全てを理解するには無職の脳ではキャパが足りな過ぎる

“

浮動小数点数とは？

- 浮動小数点数とは

- - ✓ 浮動小数点方式と呼ばれる方式で表された数のこと

● 浮動小数点数とは

✓ 浮動小数点方式と呼ばれる方式で表された数のこと

✓ 小数を **仮数部と指数部に分けて表現**

・ つまり前回やった指数表記である

$$1111.0 = \underbrace{1.111}_{\text{仮数部}} \times \underbrace{2^3}_{\text{指数部}}$$

● 浮動小数点数とは

✓ 浮動小数点方式と呼ばれる方式で表された数のこと

✓ 小数を **仮数部と指数部に分けて表現**

・ つまり前回やった指数表記である

$$1111.0 = \underbrace{1.111}_{\text{仮数部}} \times \underbrace{2^3}_{\text{基数}} \quad \text{指数部}$$

✓ 仮数部の小数点の位置が指数部によってふわ～と動くので浮動小数点と呼ぶ

● 浮動小数点数とは

✓ 浮動小数点方式と呼ばれる方式で表された数のこと

✓ 小数を **仮数部と指数部に分けて表現**

・ つまり前回やった指数表記である

$$1111.0 = \underbrace{1.111}_{\text{仮数部}} \times \underbrace{2^3}_{\text{基数}} \quad \text{指数部}$$

✓ 仮数部の小数点の位置が指数部によってふわ～と動くので浮動小数点と呼ぶ

✓ 浮動小数点数の表現方法として **IEEE754形式** が幅広く使われている

● 浮動小数点数とは

✓ 浮動小数点方式と呼ばれる方式で表された数のこと

✓ 小数を **仮数部と指数部に分けて表現**

・ つまり前回やった指数表記である

$$1111.0 = \underbrace{1.111}_{\text{仮数部}} \times \underbrace{2^3}_{\text{指数部}}$$

✓ 仮数部の小数点の位置が指数部によってふわ～と動くので浮動小数点と呼ぶ

✓ 浮動小数点数の表現形式として **IEEE754形式** が幅広く使われている

✓ 浮動ではなく固定小数点数というものもある

“

浮動小数点数型

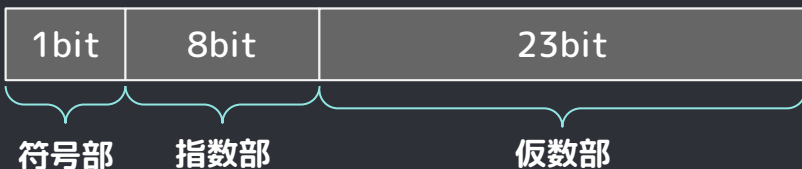
● 浮動小数点数型とIEEE754の基本形式

一般名	サイズ	基数	C言語の型
単精度	32ビット	2	float
倍精度	64ビット	2	double
四倍精度	128ビット	2	
十進倍精度	64ビット	10	
十進四倍精度	128ビット	10	

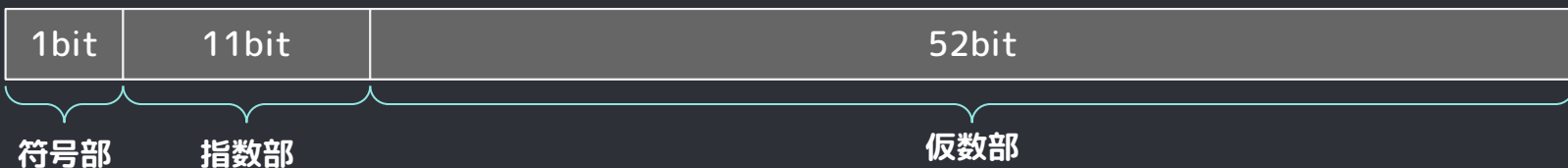
- ・ 2進数の表現だけでなく、10進数の表現も定義されている
- ・ C言語には単精度としてfloat型、倍精度としてdouble型が用意されている

浮動小数点数型 bitの内訳

float



double



型	一般名	サイズ	符号部	指数部	仮数部	最小値(約)	最大値(約)
float	単精度	32ビット	1ビット	8ビット	23ビット	-3.4e+38	3.4e+38
double	倍精度	64ビット	1ビット	11ビット	52ビット	-1.8e+308	1.8e+308

● 浮動小数点数型 指数表記との対応

$$1111.0 = 1.111 \times 2^3$$

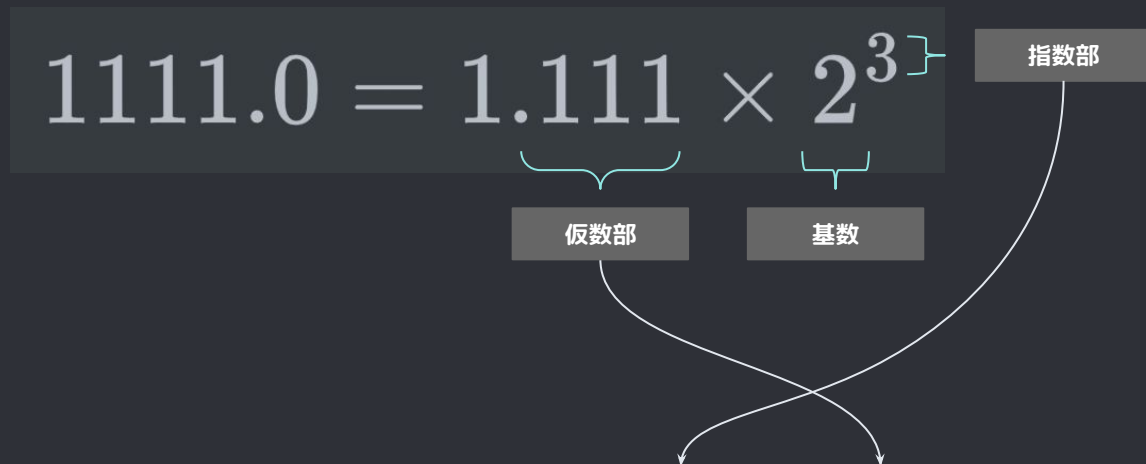
指数部

仮数部

基数

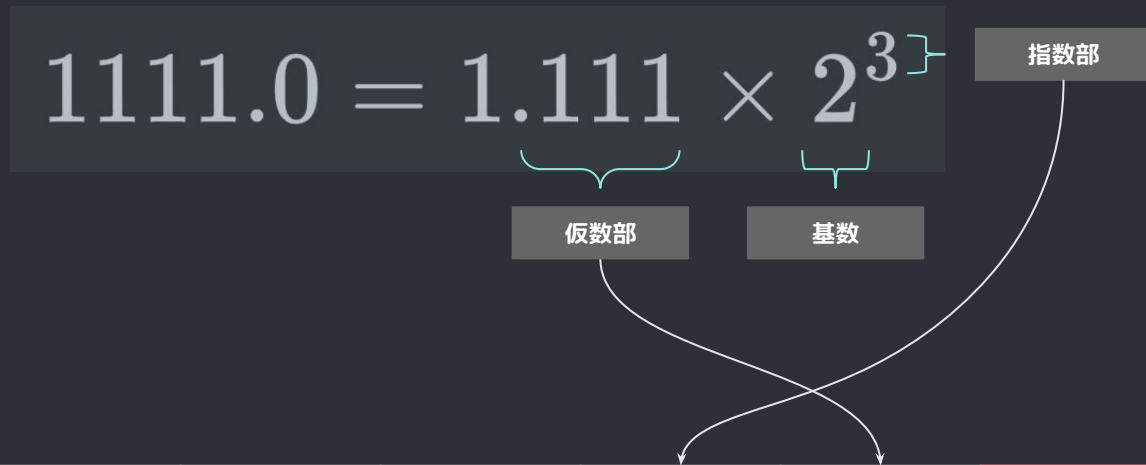
型	一般名	サイズ	符号部	指数部	仮数部	最小値(約)	最大値(約)
float	単精度	32ビット	1ビット	8ビット	23ビット	-3.4e+38	3.4e+38
double	倍精度	64ビット	1ビット	11ビット	52ビット	-1.8e+308	1.8e+308

● 浮動小数点数型 指数表記との対応



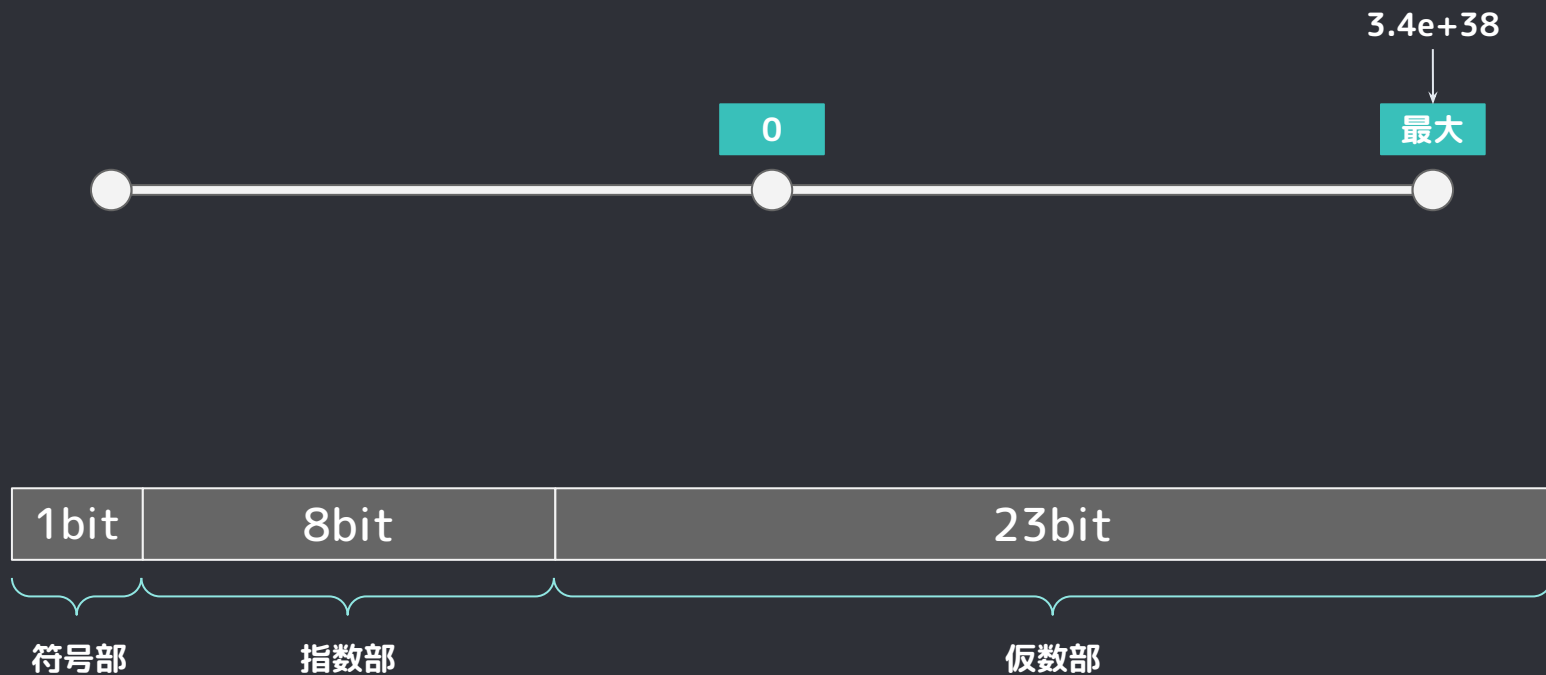
型	一般名	サイズ	符号部	指数部	仮数部	最小値(約)	最大値(約)
float	単精度	32ビット	1ビット	8ビット	23ビット	-3.4e+38	3.4e+38
double	倍精度	64ビット	1ビット	11ビット	52ビット	-1.8e+308	1.8e+308

● 浮動小数点数型 最大値と最小値

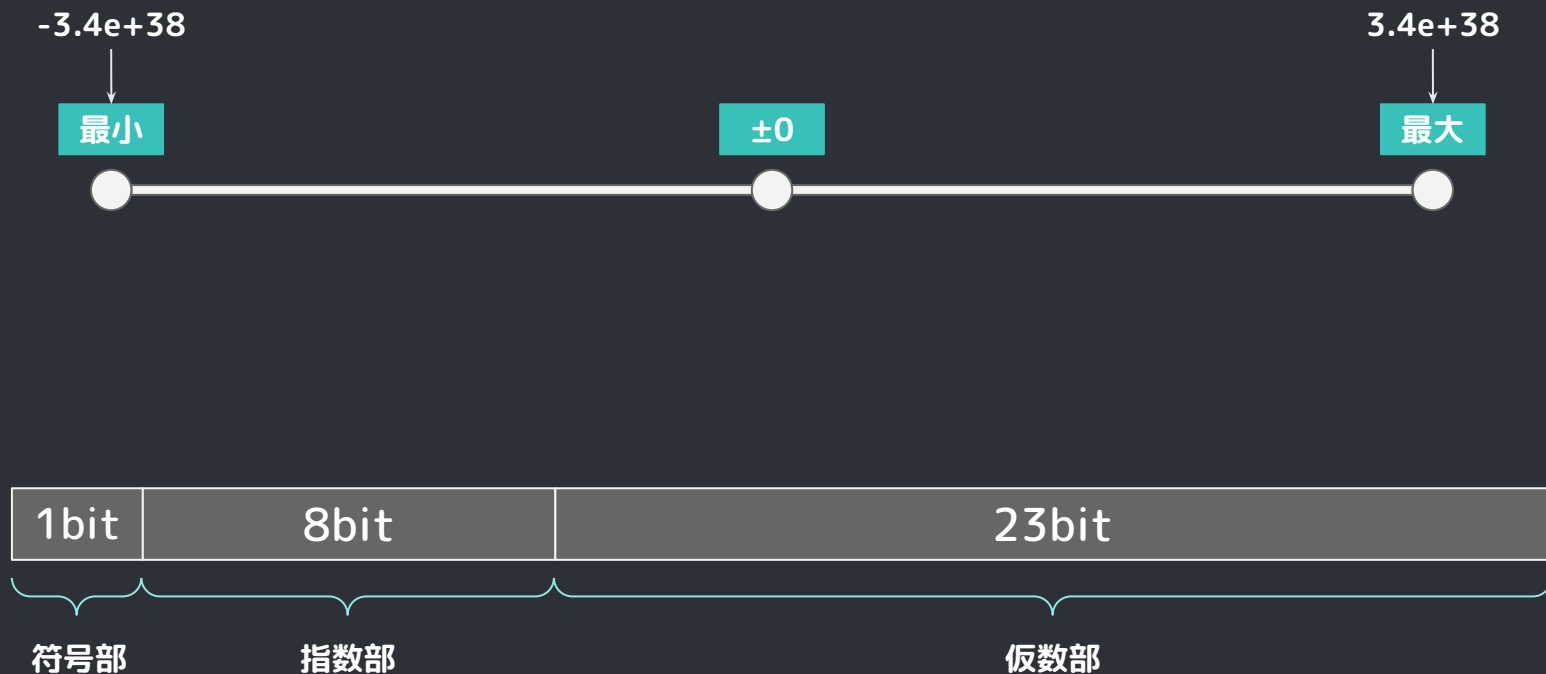


型	一般名	サイズ	符号部	指数部	仮数部	最小値(約)	最大値(約)
float	単精度	32ビット	1ビット	8ビット	23ビット	-3.4e+38	3.4e+38
double	倍精度	64ビット	1ビット	11ビット	52ビット	-1.8e+308	1.8e+308

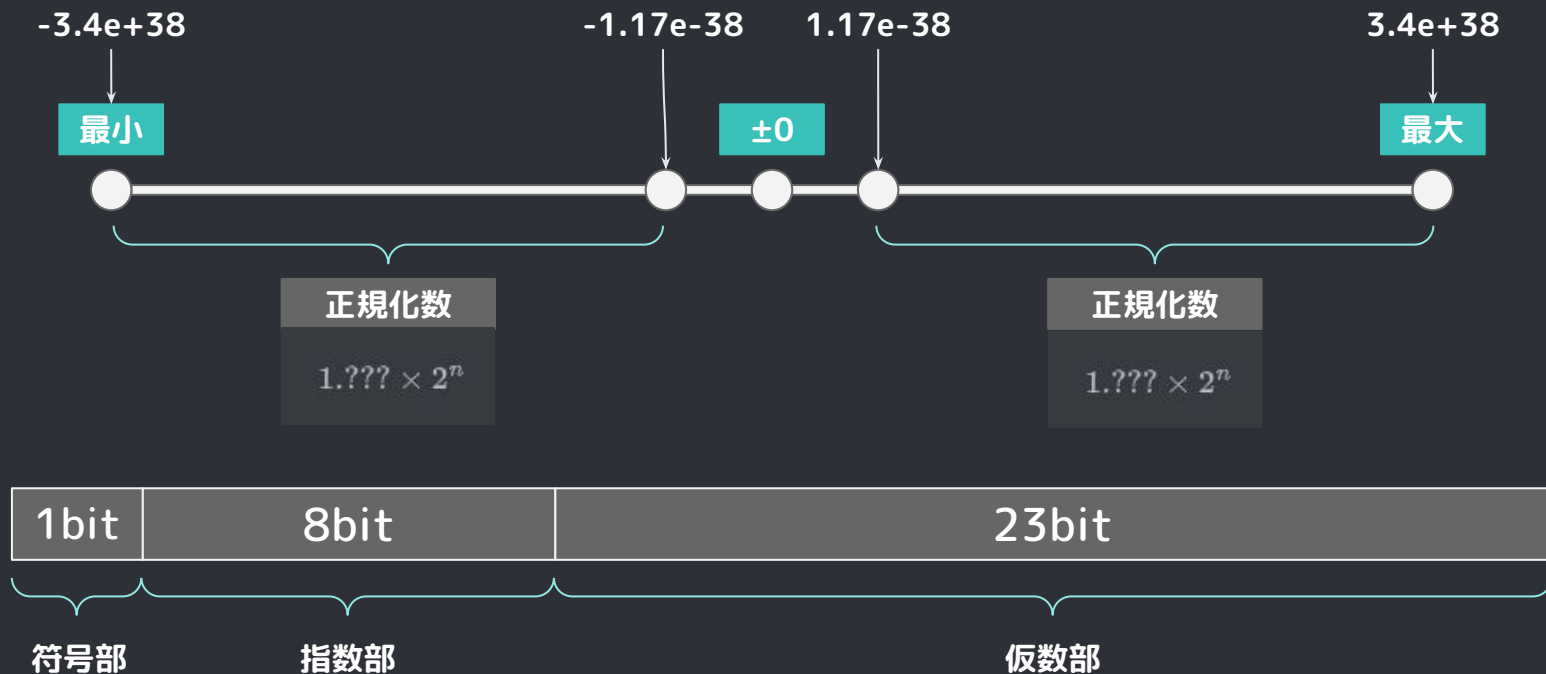
- 単精度(float)が表せるデータ



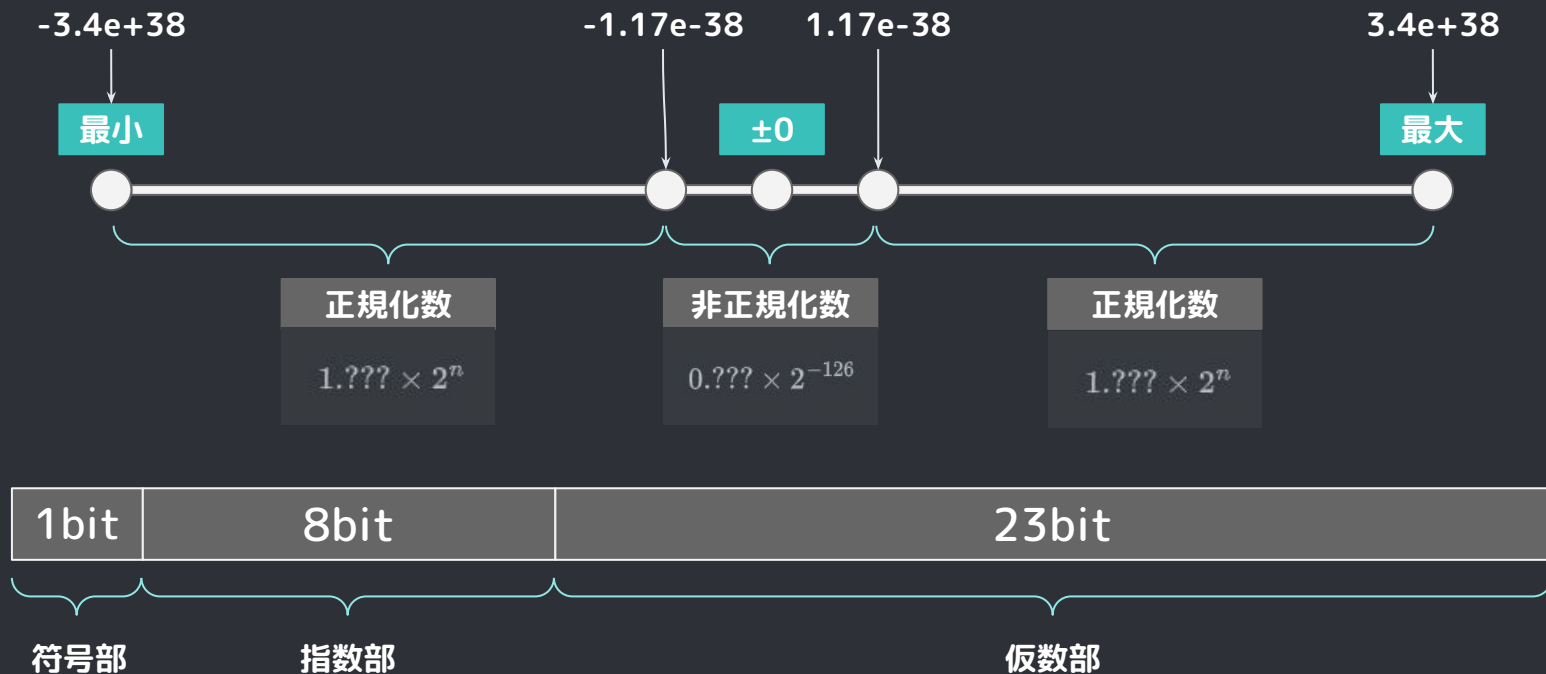
- 単精度(float)が表せるデータ



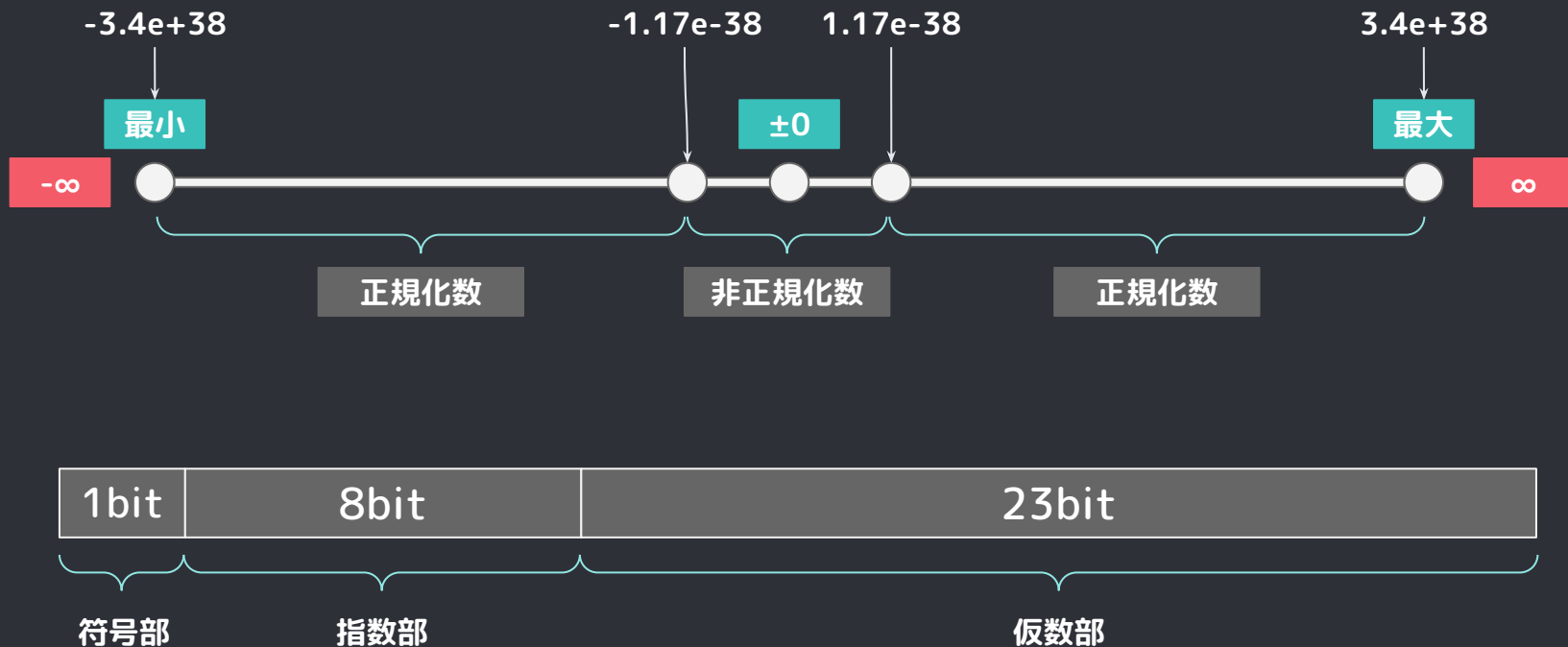
- 単精度(float)が表せるデータ



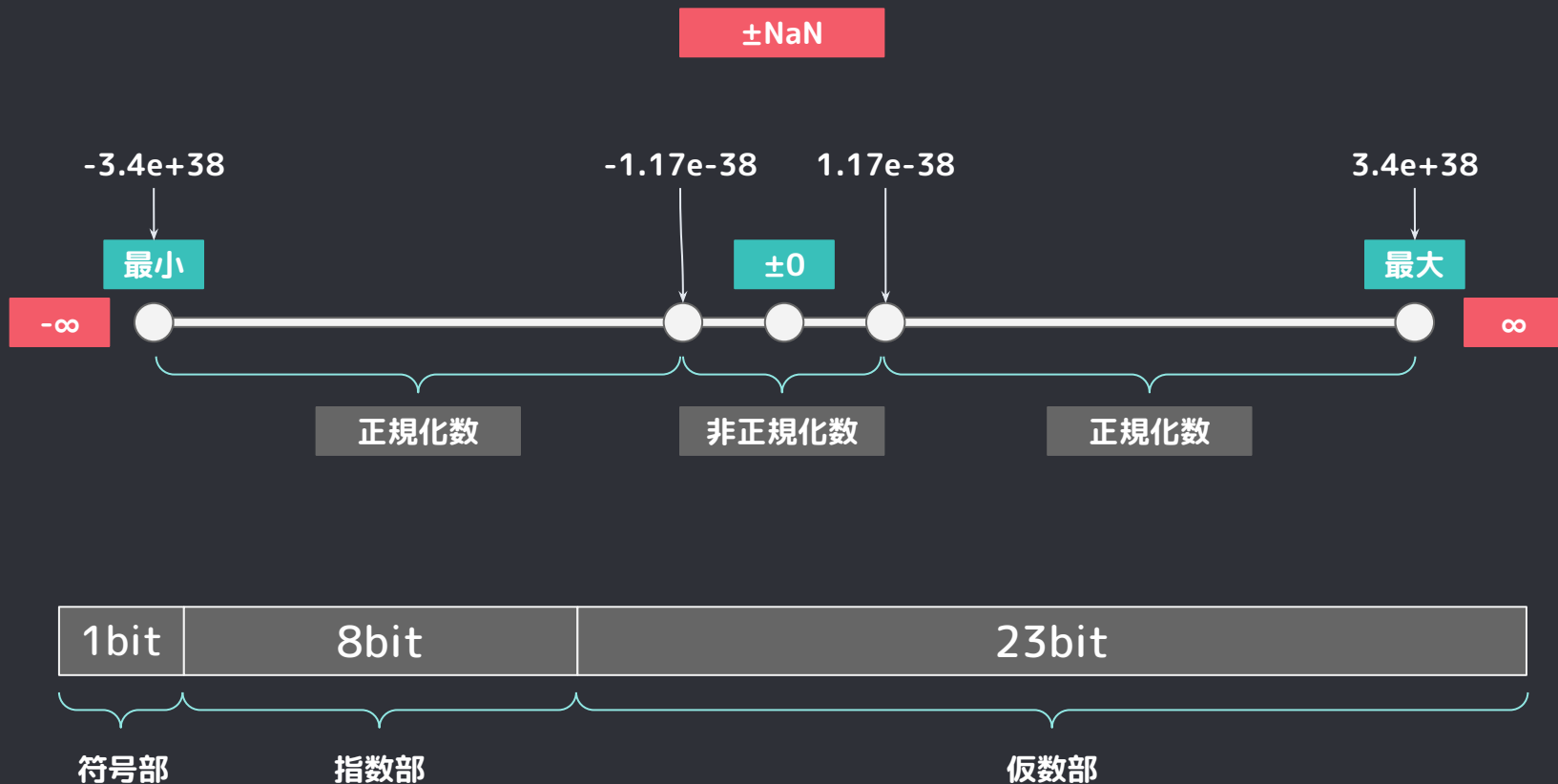
- 単精度(float)が表せるデータ



- 単精度(float)が表せるデータ



● 単精度(float)が表せるデータ



- 単精度(float)が表せるデータ

○ 浮動小数点型は以下の5つのデータを表現可能

```
float·normalized;···// 正規化数  
float·denormalized;···// 非正規化数  
float·zero;·····// ゼロ  
float·inf;·····// 無限大  
float·nan;·····// 非数
```

名前	値	種類
● normalized	1.50000000	float
● denormalized	1.401e-45#DEN	float
● zero	0.00000000	float
● inf	inf	float
● nan	nan	float

● 単精度(float)が表せるデータ

○ 浮動小数点型は以下の5つのデータを表現可能

```
float·normalized;···//正規化数  
float·denormalized;···//非正規化数  
float·zero;·····//ゼロ  
float·inf;·····//無限大  
float·nan;·····//非数
```

正規化数

$$1.??? \times 2^n$$

非正規化数

$$0.??? \times 2^n$$

仮数部が1以上、2未満なら **正規化数**

1以下の場合は **非正規化数** と呼ぶ

名前	値	種類
normalized	1.50000000	float
denormalized	1.401e-45#DEN	float
zero	0.00000000	float
inf	inf	float
nan	nan	float

“

ビットの扱い

指数部(8bit)

仮数部(23bit)

符号部(1bit)

データ

1bit

8bit

23bit

符号部

指数部

仮数部

指数部(8bit)

仮数部(23bit)

符号部(1bit)

データ

1~254

0

255

1bit

8bit

23bit

符号部

指数部

仮数部

指数部(8bit)

仮数部(23bit)

符号部(1bit)

データ

1~254

任意

0

0

0以外

255

0

0以外

1bit

8bit

23bit

符号部

指数部

仮数部

指数部(8bit)

仮数部(23bit)

符号部(1bit)

データ

1~254

任意

0

1

0

0

0

1

0以外

0

1

255

0

0

1

0以外

0

1

1bit

8bit

23bit

符号部

指数部

仮数部

指数部(8bit)

仮数部(23bit)

符号部(1bit)

データ

1~254

任意

0

1

+正規化数

-正規化数

0

0

0

1

0以外

0

1

255

0

0

1

0以外

0

1

1bit

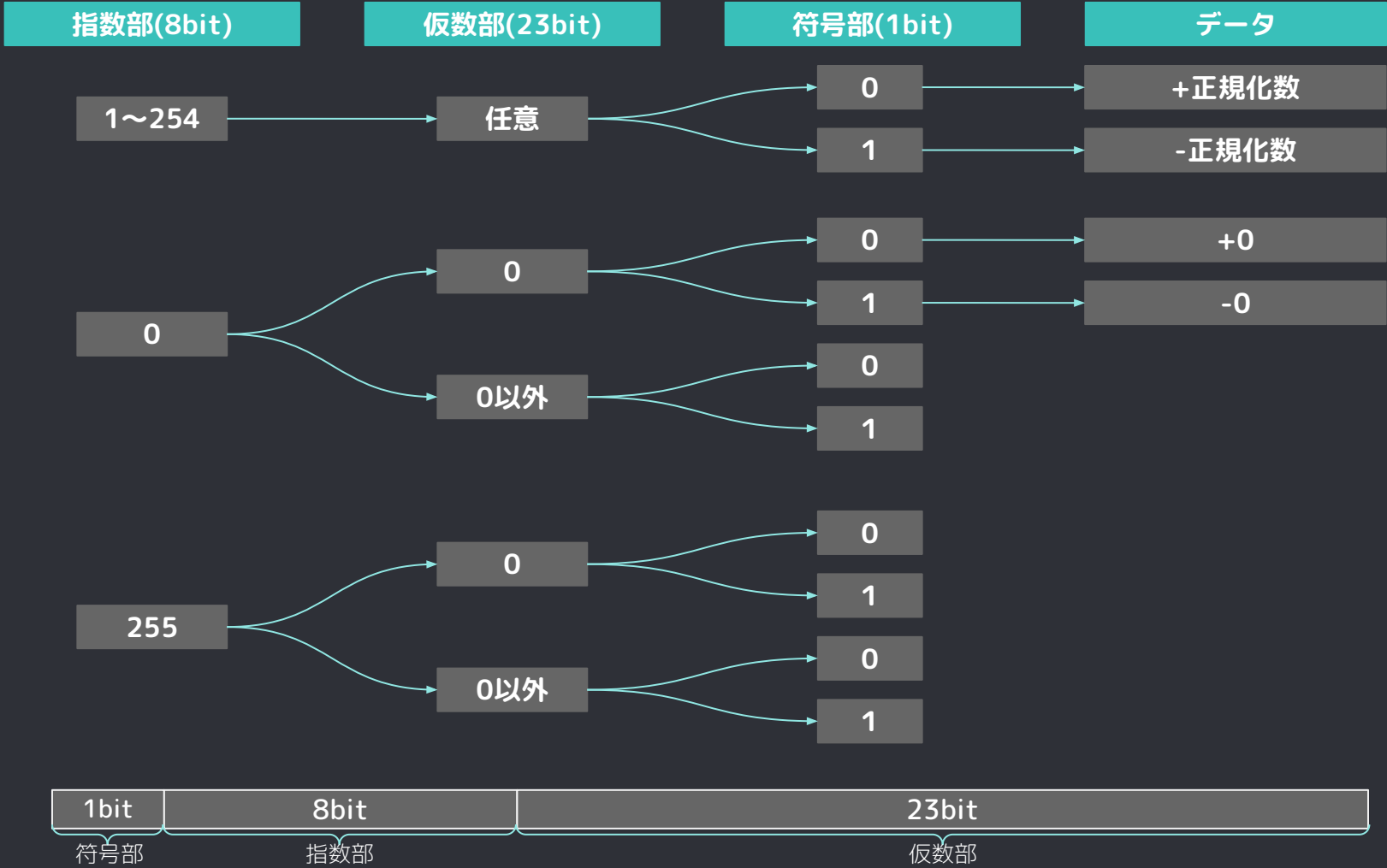
8bit

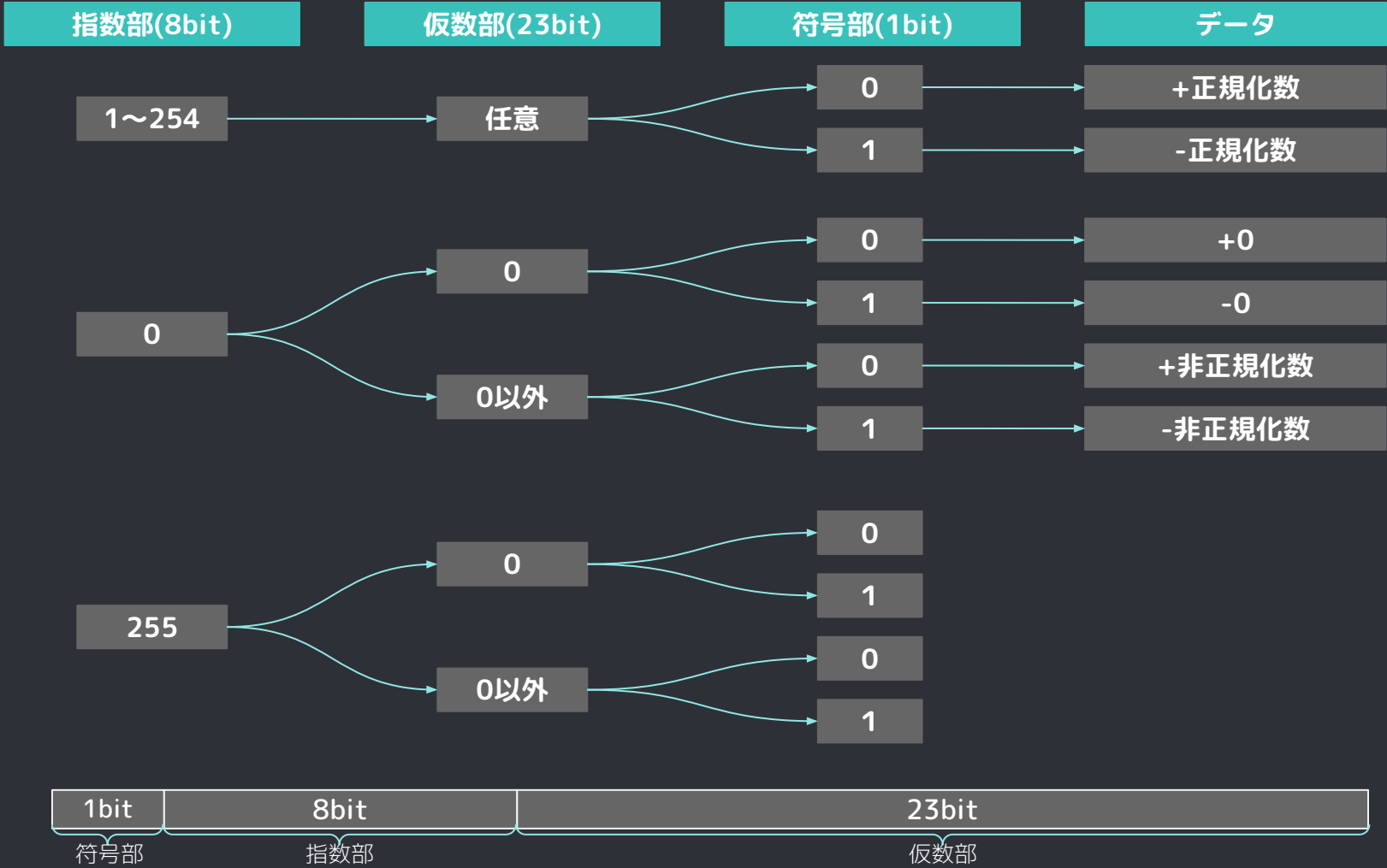
23bit

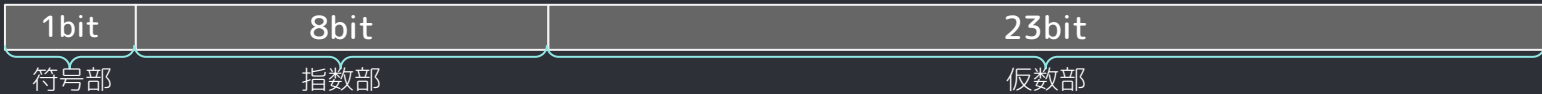
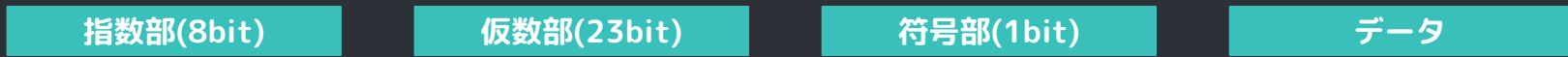
符号部

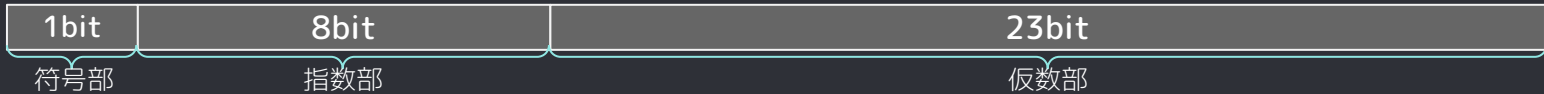
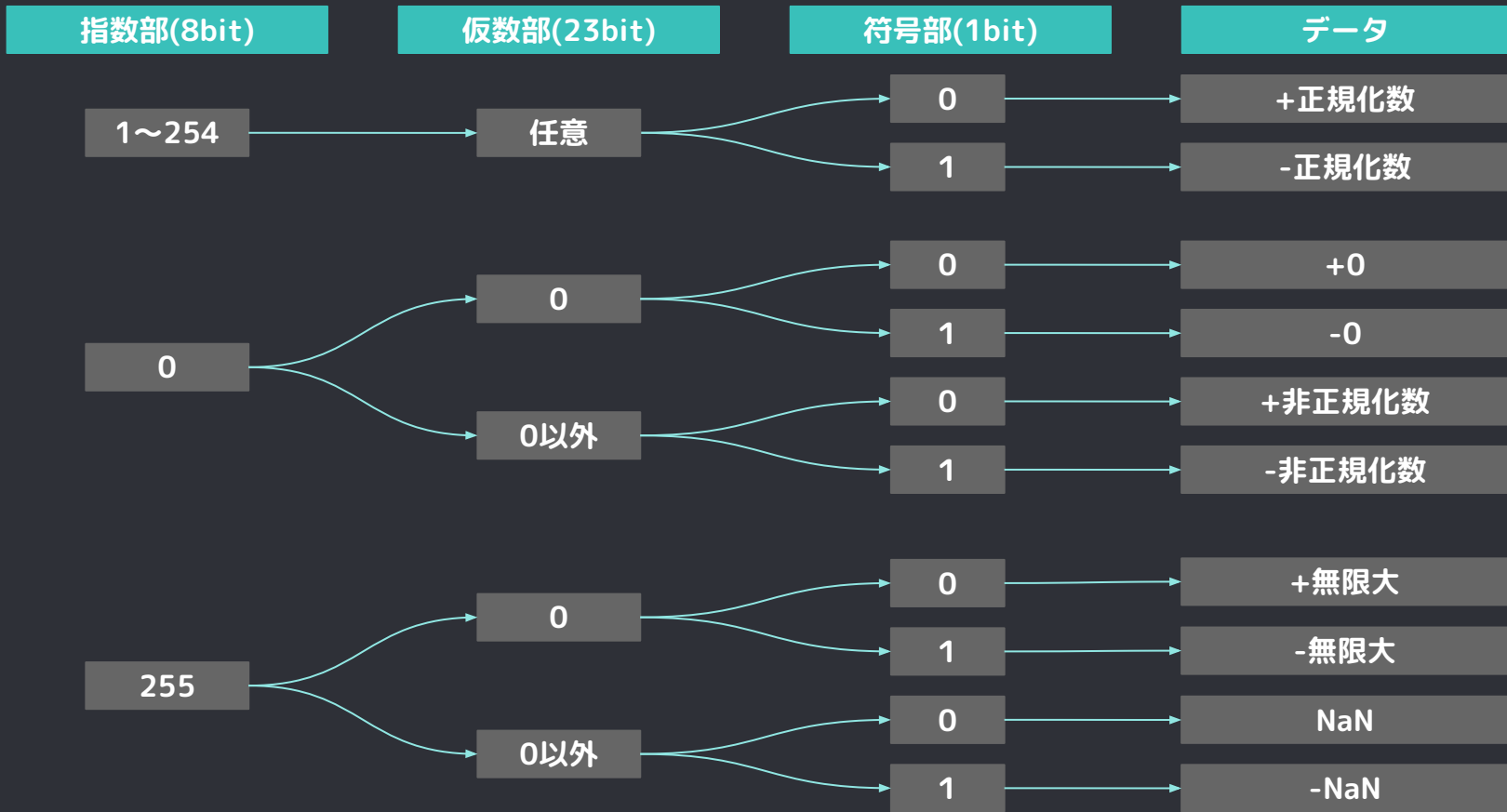
指数部

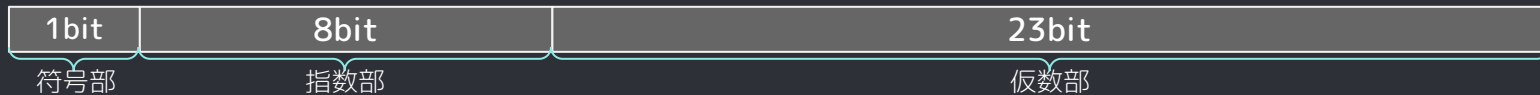
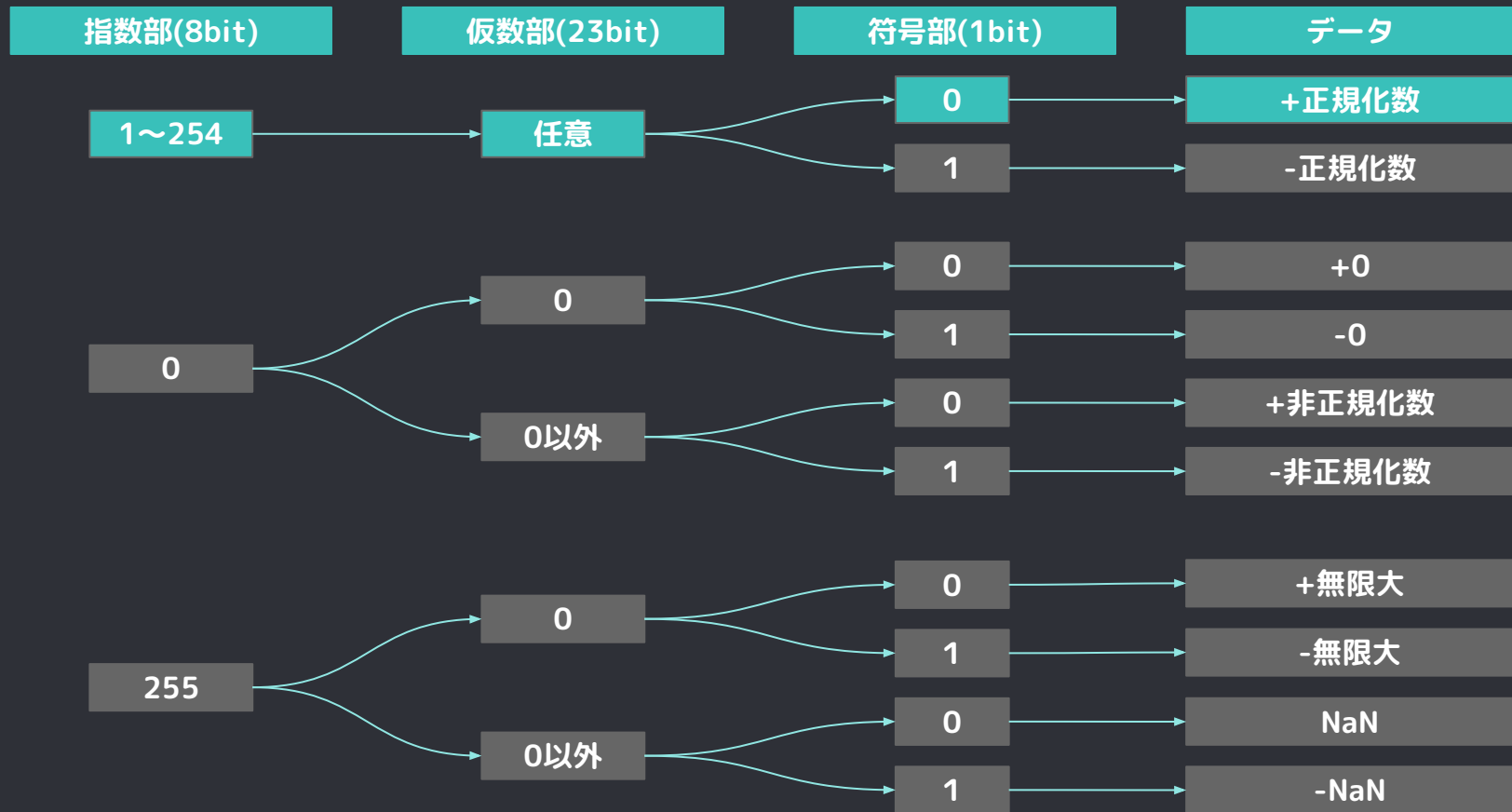
仮数部











“

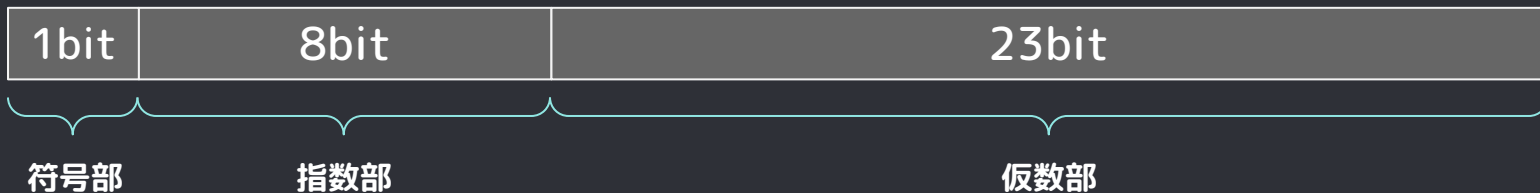
10進数の1234.0をIEEE754形式で表す

位取り記数法

1234.0

指数表記(正規化)

IEEE754形式



仮数部、数倍だー!!!



これは世を忍ぶ仮の姿

位取り記数法

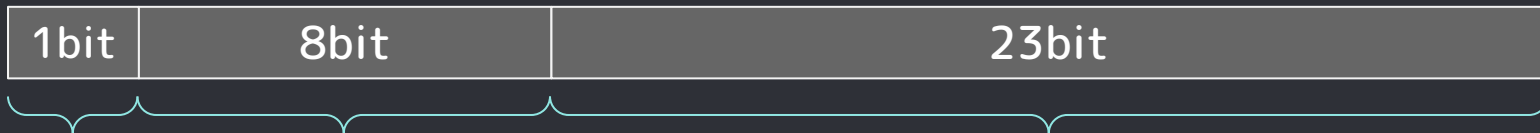
1234.0

2進数へ

10011010010.0

指数表記(正規化)

IEEE754形式



符号部



指数部



仮数部、数倍だー!!!

仮数部



これは世を忍ぶ仮の姿

位取り記数法

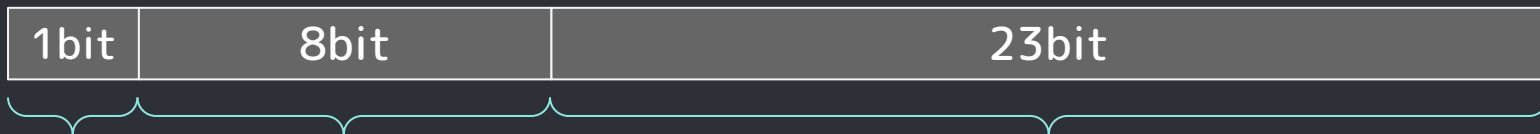
1234.0

10011010010.0

指数表記(正規化)

1.00110100100×2 の 10 乗

IEEE754形式



符号部



指数部



仮数部、数倍だー!!!

仮数部



これは世を忍ぶ仮の姿

位取り記数法

1234.0

10011010010.0

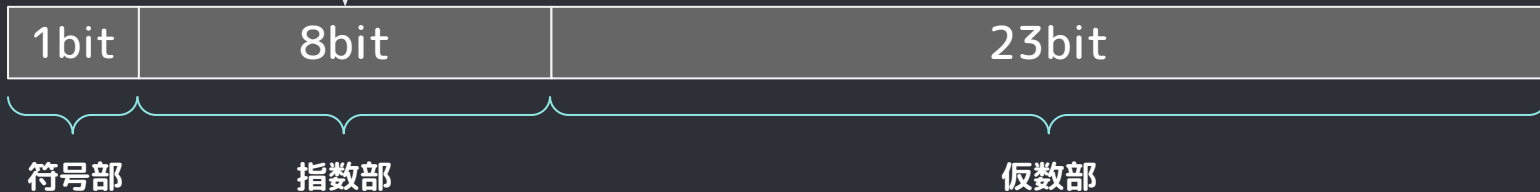
指数表記(正規化)

1.00110100100 × 2の10乗

IEEE754形式

10+127

指数部には実際の指数に127を加えた値を格納する



仮数部、数倍だー!!!



これは世を忍ぶ仮の姿

位取り記数法

1234.0

10011010010.0

指数表記(正規化)

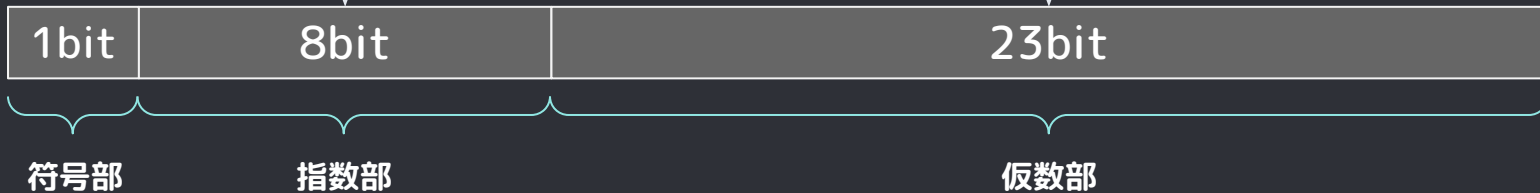
1.00110100100 × 2の10乗

正規化した値の1桁目は必ず1なので
仮数部には1.~の後ろ部分を格納する(ケチ表現)

IEEE754形式

10+127

00110100100 を23bitに0埋め



仮数部、数倍だー!!!



これは世を忍ぶ仮の姿

位取り記数法

1234.0

10011010010.0

指数表記(正規化)

1.00110100100 × 2の10乗

IEEE754形式

10+127

00110100100 を23bitに0埋め

0	1000 1001	001 1010 0100 0000 0000 0000
---	-----------	------------------------------

符号部

指数部

仮数部



仮数部、数倍だー!!!



これは世を忍ぶ仮の姿

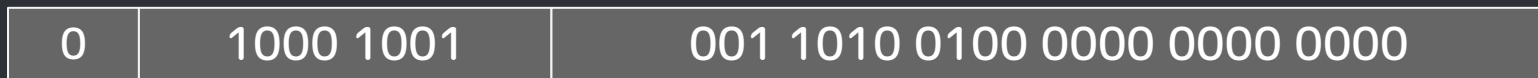
“

逆にIEEE754形式の数を10進数に戻す

位取り記数法

指数表記(正規化)

IEEE754形式



符号部

指数部

仮数部



仮数部、数倍だー!!!



これは世を忍ぶ仮の姿

位取り記数法

指数表記(正規化)

1.00110100100

IEEE754形式

1.仮数部の形にする

0	1000 1001	001 1010 0100 0000 0000 0000
---	-----------	------------------------------

符号部

指数部

仮数部



仮数部、数倍だー!!!



これは世を忍ぶ仮の姿

位取り記数法

指数表記(正規化)

1.00110100100

×

2の10乗

IEEE754形式

137-127

1.仮数部の形にする

0 1000 1001 001 1010 0100 0000 0000 0000

符号部

指数部

仮数部



仮数部、数倍だー!!!



これは世を忍ぶ仮の姿

位取り記数法

1234.0

10進数へ

10011010010.0

指数表記(正規化)

1.00110100100

×

2の10乗

IEEE754形式

137-127

1.仮数部の形にする

0	1000 1001	001 1010 0100 0000 0000 0000
---	-----------	------------------------------

符号部

指数部

仮数部



仮数部、数倍だー!!!



これは世を忍ぶ仮の姿



練習

- 次の10進数で表された小数をIEEE754形式で表せ

○ (1) 7.5

- 次の10進数で表された小数をIEEE754形式で表せ

○ (1) 7.5

位取り記数法

7.5

111.1

指数表記(正規化)

1.111 × 2の2乗

IEEE754形式

2+127

111 を23bitに0埋め

0

1000 0001

111 0000 0000 0000 0000 0000

ニブル単位

0100

0000

1111

0000

0000

0000

0000

0000

2進数

4

0

F

0

0

0

0

0

16進数

- 次の10進数で表された小数をIEEE754形式で表せ

○ (2) 2048.875

- 次の10進数で表された小数をIEEE754形式で表せ

○ (2) 2048.875

位取り記数法

2048.875

1000 0000 0000.111

指数表記(正規化)

1.000 0000 0000 111 × 2の11乗

IEEE754形式

11+127

000 0000 0000 111 を23bitに0埋め

0

1000 1010

000 0000 0000 1110 0000 0000

ニブル単位

0100

0101

0000

0000

0000

1110

0000

0000

2進数

4

5

0

0

0

E

0

0

16進数

- 次の10進数で表された小数をIEEE754形式で表せ

○ (3) 0.1

- 次の10進数で表された小数をIEEE754形式で表せ

○ (3) 0.1

位取り記数法

0.1

0.00011001100...

指数表記(正規化)

1.10011001100... × 2の-4乗

IEEE754形式

-4+127

100 11001100... を23bit分

0	0111 1011	100 1100 1100 1100 1100 1100
---	-----------	------------------------------

ニブル単位

0011

1101

1100

1100

1100

1100

1100

1100

3

D

C

C

C

C

C

C

“

正規化数以外のデータの例

● 正規化数以外のデータ例

例	bit列	16進数 (big)	16進数 (little)
0	0000 0000 0000 0000 0000 0000 0000 0000	00 00 00 00	00 00 00 00
非正規化数	0 000 0000 0 0000 0000 0000 0000 0000 0001	00 00 00 01	01 00 00 00
∞	0 111 1111 1 0000 0000 0000 0000 0000 0000	7f 80 00 00	00 00 80 7f
NaN	0 111 1111 1 0000 0000 0000 0000 0000 0001	7f 80 00 01	01 00 80 7f

- ・ 0 = 全てのbitが0
- ・ 非正規化数 = 指数部が0、仮数部が0以外
- ・ ∞ = 指数部が255、仮数部が0
- ・ NaN = 指数部が255、仮数部が0以外



おしまい