北京大学 PEKING UNIVERSITY | 物理学院

Data Lab（浮点数部分）

包涛尼

北京大学物理学院

2022 年 9 月 21 日

Contents

浮点数回顾

IEEE 754 标准

- 二进制浮点数的最高有效位被指定为符号位；「指数部分」，即次高有效的 e 个比特，存储指数部分；最后剩下的 f 个低有效位的比特，存储「有效数」（significand）的小数部分.

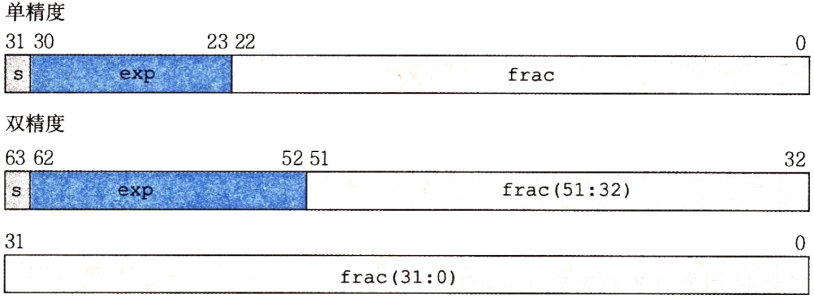

图 1 标准浮点格式

IEEE 754 标准

- IEEE 浮点数标准用 $V = (-1)^s \times M \times 2^E$ 的形式表示一个数：
- **符号**（sign）：s 决定浮点数是负数（$s = 1$）还是正数（$s = 0$），对于 0 作特殊处理.
- **尾数**（significand）：M 是一个二进制小数，其范围为 $1 \sim 2 - \varepsilon$，或者 $0 \sim 1 - \varepsilon$.
- **阶码**（exponent）：E 的作用是对浮点数加权，权重为 2^E（幂可以为负）.

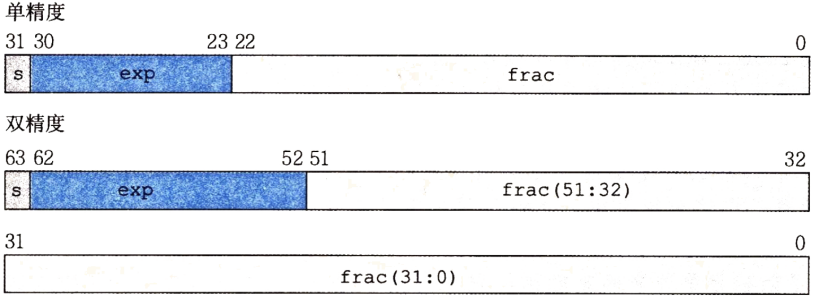

图 2 标准浮点格式

IEEE 754 标准（单精度浮点数 `float`）

- **规格化的值**（normalized）：

  $\exp \neq 0 \wedge \exp \neq 255$，$E = \exp - \mathrm{Bias}$，$\mathrm{Bias} = 2^{k-1} - 1 = 127$

  E 取值从 $-126 \sim 127$，注意尾数 `frac` 的 implied leading 1

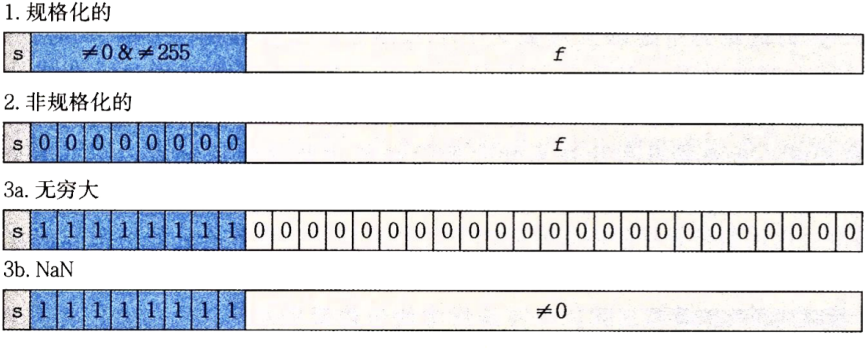

图 3　单精度浮点数分类

IEEE 754 标准（单精度浮点数 `float`）

- **非规格化的值**（denormalized）：

  $\exp = 0$，$\boxed{E = 1 - \text{Bias}}$，$\text{Bias} = 2^{k-1} - 1 = 127$

  $E = -126$ 为定值，注意尾数 `frac` 的 implied leading 0

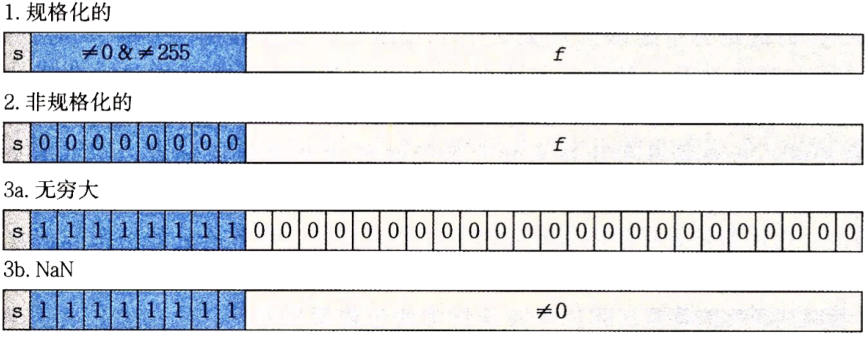

图 4　单精度浮点数分类

IEEE 754 标准（单精度浮点数 `float`）

- **特殊值**（special）：

  $\texttt{exp} = 255$

  $\texttt{frac} = 0$ 时代表无穷大，$\texttt{frac} \neq 0$ 时代表 NaN（Not a Number）

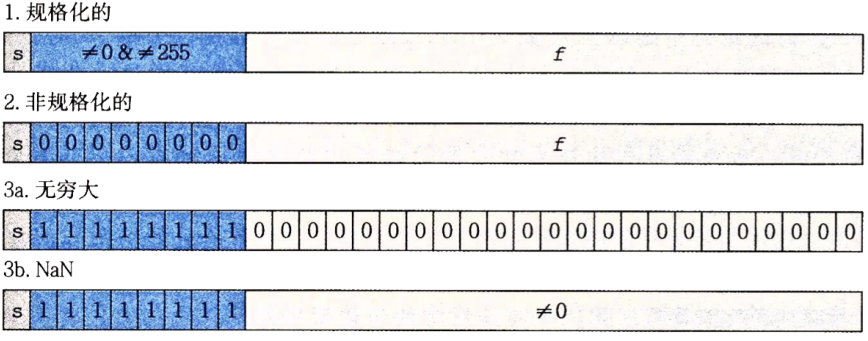

图 5　单精度浮点数分类

Data Lab 浮点数部分

浮点数题目的说明与要求

可以的：

- 可以使用**循环**语句和**条件控制**语句.
- 可以使用**整型**（`int`）和**无符号整型**（`unsigned`）及相关操作.
- 可以使用任意的整型/无符号整型的常量.

不可以的：

- 不得调用任何函数.
- 不得使用 `int` 或 `unsigned` 以外的任何数据类型（包括数组、结构体或联合体）.
- 不得使用浮点型（`float`）及相关操作或常量.

输入和输出都应该是 `unsigned`，不过应被解释为单精度浮点数 `float` 的比特级表示.

第一题：float_twice

题目要求：Return bit–level equivalent of expression $2 * f$ for floating point argument `f`.

```
1  unsigned float_twice(unsigned uf)
2  {
3      unsigned exp = (uf >> 23) & 0x000000ff;
4      unsigned sign = uf & (1 << 31);
5      if (exp == 0)
6          return uf << 1 | sign;
7      if (exp == 255)
8          return uf;
9      exp = exp + 1;
10     if (exp == 255)
11         return 0x7f800000 | sign;
12     return sign | exp << 23 | (uf & 0x007fffff);
13 }
```

第二题：float_i2f

题目要求：Return bit–level equivalent of expression (float) x.

```
1  unsigned float_i2f(int x)
2  {
3      int f;
4      int sign_x = x & 0x80000000;
5      int e = 0x0000009e;
6      if (x == 0x80000000)
7          return 0x80000000 | (158 << 23);
8      if (!x)
9          return 0;
10     if (sign_x)
11         x = ~x + 1;
12     while (!(x & 0x80000000))
13     {
14         x = x << 1;
15         e = e - 1;
16     }
17     f = ((x & (~0x80000000)) >> 8);
18     if (((x & 0x0000007f) > 0 || f & 1) &&
           (x & 0x00000080))
19         f = f + 1;
20     return sign_x + f + (e << 23);
21 }
```

第三题：float_f2i

题目要求：Return bit-level equivalent of expression (int) f.

```
1  int float_f2i(unsigned uf)
2  {
3      int exp = (uf >> 23) & 0x000000ff;
4      int frc = (uf & 0x007fffff);
5      int bias = 0x0000007f;
6      int res = frc;
7      if (exp == 0x000000ff)
8          return 0x80000000u;
9      if (exp < bias)
10         return 0;
11     exp = exp - bias;
12     if (exp >= 0x0000001f)
13         return 0x80000000u;
14     if (exp > 0x00000016)
15         res = frc << (exp - 0x00000017);
16     else
17         res = frc >> (0x00000017 - exp);
18     res = res + (1 << exp);
19     if (uf >> 31)
20         res = -res;
21     return res;
22 }
```

第四题：float_pwr2

题目要求：Return bit–level equivalent of the expression 2^x for any 32–bit integer x.

```
1  unsigned float_pwr2(int x)
2  {
3      if (x < -149)
4          return 0;
5      if (x < -126)
6          return 1 << (149 + x);
7      if (x < 128)
8          return (x + 127) << 23;
9      return 0x7f800000;
10 }
```

参考文献

[1] Bryant R E, O'Hallaron D R. 深入理解计算机系统 [M]. 北京: 机械工业出版社, 2016.

谢 谢！