国际 IEEE 754 标准，为啥会有精度缺失

作者：kakj-go
原文链接：https://ld246.com/article/1571151420867
来源网站：链滴

国际标准IEEE 754 浮点数的问题

- 整数的二进制比如 2 的二进制是10没疑问

- 小数的二进制比如 0.125 的二进制是 001 流程如下

//每次乘以2取整数部分，每次取完整数部分保留小数部分再乘以2

0.125

0.25=0.125*2 0

0.5=0.25*2 0

1.0=0.5*2 1

0.0=0.0*2 0

0.0 0

问题

1. 为啥这样存有什么原因，有什么好处和坏处

2. 为啥要指数表达法

3. 为啥要 补位

4. 为啥忽略 1

好了，上面这些问题，没事就是让你知道有这种问题就行，因为希望你在看下面的例子会产生这样的问，最后会一一解答

IEEE 754的内存结构

| 数符 | 阶码 | 尾数 |

| ---- | ---- | ----|

| 1位 | 8位 | 23位 |

整体的结构就是

0 00000000 00000000000000000000000 (问题1)

2.125的内存二进制表达

1. 整数位 2 获取到它的二进制为 10

2. 0.125 的二进制表达为 001

3. 合并2个二进制就是 10.001

4. 二进制的指数表达法(问题2)为 1.0001 * 2^1 因为小数点左移了 1 位所以是 2^1

5. 补位 127 (问题3) 1+127=128 128二进制为 10000000, 1 是 2^1 中的

6. 整体的二进制表示就是 0 10000000 00010000000000000000000

其中数符 0 代表是正数，对应1代表负数, 阶码 10000000 是第五步产出的, 尾数 00010000000000000000000 是第四步的 1.0001 去除整数 1(问题4) 后补位到23位产出的

0.001的内存二进制表达

1. 整数位 0 获取二进制为 0

2. 0.001 获取二进制为 0000000001000001100010010011011111111111111111...

3. 合并就是 0.0000000001000001100010010011011111111111111111...

4. 二进制的指数和表达法为 1.00000110001001001101111111111111111... * 2^-10 因为向右移动10位所以是 2^-10

5. 补位 -10+127=117 二进制为=1110101, 注意这里是7位，下面会补0

6. 整体的二进制表示就是 0 01110101 00000110001001001101111111111111111

内存结构同2.125分析过程

个人理解

1. 为啥这样存有什么原因，有什么好处和坏处

- 首先假如我们不这样存, 2.125 我们分成整数位和小数位分别存到内存中，那么一个整数位大小就能用32位，小数位占用32位，总共一个float就占用64位

- 其次0.001这种用二进制表示如何表示,只能00000000010000011000100100110111111111111111...,这样用float依然会产生精度缺失的问题, 还有就是假如 0000000.....000001 这种情况怎么办，也就前面的0就占了很多位，后面的数字根本存不了多大

- 好处就是上面的坏处, 占用内存小，精度依然缺失，可以存储这种很长0的数

2. 为啥去除指数表达式中的 1 (1.xxxxx)去除1留下xxxxx，首先我们大致知道了二进制指数表达法，本质就是为了去除小数部分的0开头的数据，因为0全部保留那么就像解答1中所说的那样0怎么去存储这里的二进制指数表达式就很优雅的去除了0，用的就是2^x来代表0,然后把1.xxx中的1去除，因为一使用了二进制指数表达式，你的最前面一定是个1，这也是这个1为啥可以去除

，这样x就可以代表偏移去除的0值，就可以单独代表很多小数

3. 上面2个例子的补位是啥操作+127

目前不知道

4. 忽略1是因为前面都是1所以可以少存一位

总结

总的来讲，你要体会为啥要使用二进制的指数表达法，首先是为了去除整数和小数之间的区分，整体成小数，然后用阶码来表示是变大了还是变小了，用尾数代表真正的数，然后要体会为啥是这样存储本质就是为了存储二进制指数表达法

关联博客阅读:

1. https://www.cnblogs.com/backwords/p/9826773.html

2. https://blog.csdn.net/fwb330198372/article/details/70238982

工具:

1. https://tool.lu/hexconvert/ 二进制转10进制

2. http://www.binaryconvert.com/result_float.html?decimal=048046048048049 浮点转二进制