Representación Flotante

Floating representation

Samuel Martínez Cruz
Universidad Tecnológica de Pereira
Samuel.martinez@utp.edu.co

***Resumen*— Este documento contiene un resumen sobre la representación flotante o punto flotante, en el campo de informática. El objetivo es realizar una revisión de los números flotantes en las computadoras.**

Palabras clave— número, exponente, mantisa, signo, bits , aritmética

***Abstract*—**
This document contains a summary on the floating representation or floating point, in the computer field. The goal is to perform a review of floating numbers on computers.

***Key Word* — number, exponent, mantissa, sign, bits, arithmetic**

I. INTRODUCCIÓN

La representación de números con punto flotante es muy común observarlas en la informática, pues las computadoras no pueden operar con números demasiados grandes, por lo que deben reducir esa cantidad a una mucho más pequeña para de este modo poder operarlas. En este documento se encontrará la definición de lo que significa en informática los números flotantes, cómo se pasa un número demasiado grande a uno más pequeño, para qué son útiles y cómo funcionan.

II. CONTENIDO

¿Qué es la representación de punto flotante?

La representación de punto flotante (en inglés floating point) es una forma de notación científica usada en los computadores con la cual se pueden representar números reales extremadamente grandes y pequeños de una manera muy eficiente y compacta, y con la que se pueden realizar operaciones aritméticas. El estándar actual para la representación en coma flotante es el IEEE 754. [1]

¿Por qué son necesarios los números con punto flotante?

Como la memoria de los ordenadores es limitada, no puedes almacenar números con precisión infinita, no importa si usas fracciones binarias o decimales: en algún momento tienes que cortar. Pero ¿cuánta precisión se necesita? ¿Y dónde se necesita? ¿Cuántos dígitos enteros y cuántos fraccionarios?

Para un ingeniero construyendo una autopista, no importa si tiene 10 metros o 10.0001 metros de ancho ─ posiblemente ni siquiera sus mediciones eran así de precisas.
Para alguien diseñando un microchip, 0.0001 metros (la décima parte de un milímetro) es una diferencia enorme ─ pero nunca tendrá que manejar distancias mayores de 0.1 metros.
Un físico necesita usar la velocidad de la luz (más o menos 300000000) y la constante de gravitación universal (más o menos 0.0000000000667) juntas en el mismo cálculo. [2]

¿Cómo funcionan los números de punto flotante?

La idea es descomponer el número el número en dos partes:

Una **mantisa** (también llamada coeficiente o significando) que contiene los dígitos del número. Mantisas negativas representan números negativos.

Un **exponente** que indica dónde se coloca el punto decimal (o binario) en relación al inicio de la mantisa. Exponentes negativos representan números menores que uno.

Este formato cumple todos los requisitos:

Puede representar números de órdenes de magnitud enormemente dispares (limitado por la longitud del exponente).

Proporciona la misma precisión relativa para todos los órdenes (limitado por la longitud de la mantisa).

Permite Cálculos entre magnitudes: multiplicar un número muy grande y uno muy pequeño conserva la precisión de ambos en el resultado.

Los números de coma flotante decimales normalmente se expresan en notación científica con un punto explicito siempre entre el primer y el segundo dígito. El exponente o bien se escribe explícitamente incluyendo la base, o sea una **e** para separarlo de la mantisa. Así:

Mantisa	**Exponente**	**Notación científica**	**Valor en punto fijo**
1.5	4	$1.5 * 10^4$	15000
-2.001	2	$-2.001* 10^2$	-200.1
5	-3	$5 * 10^{-3}$	0.005
6.6667	-11	$6.667 * 10^{-11}$	0.0000000000677

[2]

El estándar

Casi todo el hardware y lenguajes de programación utilizan números de pinto flotante en los formatos binarios, que están definidos en el estándar IEEE 754. Los formatos más comunes son de 32 o 64 bits de longitud total:

FORMATO	**BITS TOTALES**	**BITS SIGNIFICATIVOS**
Precisión sencilla	32	23 + 1 signo
Precisión doble	64	53 + 1 signo
	BITS DEL EXPONENTE	**NÚMERO MÁS PEQUEÑO**
Precisión sencilla	8	$\sim 1.2 * 10^{-38}$
Precisión doble	11	$\sim 5.0 * 10^{-324}$
	NÚMERO MÁS GRANDE	
Precisión sencilla	$\sim 3.4 * 10^{38}$	
Precisión doble	$\sim 1.8 * 10^{308}$	

Hay algunas peculiaridades:

La secuencia de bits es primero el bit del signo, seguido del exponente y finalmente los bits significativos.

El exponente no tiene signo; en su lugar se le resta un desplazamiento (127 para sencilla y 1023 para

doble precisión). Esto, junto con la secuencia de bits, permite que los números de punto flotante se puedan comparar y ordenar correctamente incluso cuando se interpretan como enteros.

Se asume que le bit más significativo de la mantisa es 1 y se omite, excepto para casos especiales.

Hay valores diferentes a cero positivo y cero negativos. Estos difieren en el bit del signo, mientras que todos los demás son 0. Deben ser considerados iguales, aunque sus secuencias de bits sean diferentes.

Hay valores especiales **no numéricos** (NaN, <<not a number>> en inglés) en los que el exponente es todo unos y la mantisa no es todo ceros. Estos valores representan el resultado de algunas operaciones indefinidas (como multiplicar por 0 por infinito, operaciones que involucren NaN, o casos específicos). Incluso con valores NaN con idéntica secuencia de bits no deben ser considerados iguales. [2]

Representación

Representar los siguientes números en 8 bits

+ 476382496102

0	4	7	6	3	8	1	2

La primera casilla es dónde se debe indicar el signo del número, si es positivo se representa con el número 0, si es negativo con el número 1

Las dos últimas casillas indican el exponente de la base, en este caso estamos en sistema decimal entonces la base es 10

El resto de casillas es donde va ubicada la mantisa, es decir el número.

Representar: +982641367214932

0	9	8	2	6	4	1	5

Representar: -72462

1	7	2	4	6	2	0	5

Aritmética con punto flotante

Suma

Para sumar dos números de punto flotante, los exponentes deben ser iguales. Si ellos, no son iguales, entonces se deben hacer iguales, desplazando la mantisa del número con el exponente más pequeño. Por ejemplo, considere 10, 375 + 6,34375 = 16,71875 o en binario:

$$\begin{array}{rr} & 1{,}0100110 \times 2^3 \\ + & 1{,}1001011 \times 2^2 \\ \hline \end{array}$$

Estos dos números no tienen el mismo exponente así que se desplaza la mantisa para hacer iguales los exponentes y entonces sumar:

$$\begin{array}{rr} & 1{,}0100110 \times 2^3 \\ + & 0{,}1100110 \times 2^3 \\ \hline & 10{,}0001100 \times 2^3 \end{array}$$

Observe que el desplazamiento de 1,1001011 * 2^2 pierde el uno delantero y luego de redondear el resultado se convierte en 0,1100110 * 2^3. El resultado de la suma, 10,0001100 * 2^3 (ó 1,00001100 * 2^4) es igual 10000,1102 ó 16.75. Esto no es igual a la respuesta exacta (16.71875) Es sólo una aproximación debido al error del redondeo del proceso de la suma.

Es importante tener en cuenta que la aritmética de punto flotante en un computador (o calculadora) es siempre una aproximación. Las leyes de las matemáticas no siempre funcionan con número de punto flotante en un computador. Las matemáticas asumen una precision infinita que un computador

no puede alcanzar. Por ejemplo, las matemáticas enseñan que (a+b)-b = a; sin embargo, esto puede ser exactamente cierto en un computador.

Resta

La resta trabaja muy similar y tiene los mismos problemas que la suma

Considere como un ejemplo 16,75 – 15,9375 = 0,8125

$$\begin{array}{r} 1{,}0000110 \times 2^4 \\ -\ 1{,}1111111 \times 2^3 \\ \hline \end{array}$$

Desplazando $1{,}1111111 \times 2^3$ da (redondeando arriba) $1{,}0000000 \times 2^4$

$$\begin{array}{r} 1{,}0000110 \times 2^4 \\ -\ 1{,}0000000 \times 2^4 \\ \hline 0{,}0000110 \times 2^4 \end{array}$$

Multiplicación y División

Para la multiplicación, las mantisas son multiplicadas y los exponentes son sumados

Considere 10,357 * 2,5 = 25,9375

$$\begin{array}{r} 1{,}0100110 \times 2^3 \\ \times\ 1{,}0100000 \times 2^1 \\ \hline 10100110 \\ +\ \ 10100110 \\ \hline 1{,}10011111000000 \times 2^4 \end{array}$$

Claro está, el resultado real podría ser redondeado a 8 bits para dar:

1,1010000 * 2,4 = 11010,0002 = 26

La División es más compleja, pero tiene problemas similares con errores de redondeo. [3]

III. CONCLUSIONES

La representación de números con punto flotante en informática es demasiado importante, pues los computadores no tienen la capacidad para hacer operaciones aritméticas con números demasiados grandes o demasiados pequeños, y gracias a esta representación la máquina puede realizarlo, no tan exactamente, pero de algún modo puede realizarlo haciendo redondeos.

Con esta representación y con todo el contenido de este documento se puede deducir que, si o si se va a perder cierta cantidad de un valor, es decir que el resultado de determinada operación no va a ser exacta.

Los números con los que la computadora opera pueden estar dados en diferentes bases numéricas, sea octal (0 a 7), sea decimal (0 a 9), sea binaria, (0 a 1), o sea hexadecimal (0 a 15).

REFERENCIAS

[1] https://es.wikipedia.org/wiki/Coma_flotante

[2] http://puntoflotante.org/formats/fp/

[3] http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro20/39_aritmtica_de_punto_flotante.html