Table of Contents

[Khảo sát sự sai số 8](#_Toc65283143)

OVERVIEW

**Sơ lược về số thực.**

* Số thực là một tập hợp các số vô tỉ và hữu tỉ, bao gồm cả số dương, số 0 và số âm.
* Tính liên tục và trù mật của số thực.
* Xét tập hợp số hữu tỉ .

**Sự chuẩn hoá**

Với là cơ số, p là số lượng các chữ số có thể biểu diễn, sự chuẩn hoá được định nghĩa như sau:

Biểu diễn còn có thể được viết lại dưới dạng:

**Ví dụ:**

|  |  |  |
| --- | --- | --- |
| Số chưa chuẩn hoá: | Số đã chuẩn hoá | Cách diễn đạt khác |
|  |  |  |
|  |  |  |

Kiến trúc dấu chấm động theo chuẩn IEEE-754

|  |  |  |
| --- | --- | --- |
| Sign (S) | Exponent (E) | Fraction (F) |

* S: bit dùng xác định dấu (0: dương, 1: âm).
* E: phần số mũ Exponent.
* Fraction: Phần định trị

Phần định trị được xác định:

Công thức chung biểu diễn giá trị số dấu chấm động:

* bias (độ lệch): được tính bởi vớik là số lượng bit trong vùng E.
* p (độ chính xác – precession): số lượng bit có trong vùng fraction.
* : giá trị bit tại vị trí {\displaystyle {\frac {s}{b^{\,p-1}}}\times b^{e},}.

***Lưu ý:*** Do phần fraction ở hệ nhị phân ở dạng normalized luôn là 1.

Do đó:

1. Giá trị 1 trước dấu chấm trong dãy nhị phân được hiểu là 1 mà không cần phải lưu trữ.
2. Điều này có nghĩa là với p = 24(Single) ta có số bit trong vùng fraction sẽ là p – 1 = 24 – 1 = 23 (Tương tự với p = 53(Double) ta sẽ có 53 – 1 = 52 bit).

|  |  |
| --- | --- |
| **Single Precesion** | **Double Precesion** |
| 1bit Sign + 8bit Exponent + 23bit Fraction(p = 24) | 1bit Sign + 11bit Exponent + 52bit Fraction(p = 53) |
| |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 8 | 7 | … | 1 | 24 | 22 | … | 2 | 1 | |  |  | | | |  | | | | | | |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 11 | 10 | … | 1 | 53 | 50 | … | 2 | 1 | |  |  | | | |  | | | | | |

BIỂU DIỄN DẤU CHẤM ĐỘNG

Để đơn giản ta sử dụng định dạng Single Precession (cơ chế của định dạng Double Precession cũng hoạt động tương tự).

**Cách chuyển từ hệ 2 sang hệ 10 (số 32bit).**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | … | 0 |

|  |  |
| --- | --- |
| ) | |
|  |  |
|  |  |

Vậy kết quả khi chuyển từ hệ nhị phân sang hệ 10

Cách chuyển từ hệ 10 sang hệ 2

|  |  |  |
| --- | --- | --- |
| Phần Dấu | Phần Nguyên | Phần Thập Phân |
| 1(S = 0) |  |  |

* Tìm dãy số nhị phân cho phần nguyên và phần thập phân
  + Phần Nguyên:
  + Phần Thập Phân:

|  |  |  |
| --- | --- | --- |
|  |  | 1 |
|  |  | 1 |
|  |  | 0 |
|  | (Dừng) | 1 |

* + Gộp Phần Nguyên và Phần Thập Phân lại:
* Normalize kết quả:
* Tính toán số mũ
  + Ta có phần mũ sau khi chuẩn hoá là

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

Vậy kết quả biểu diễn nhị phân của

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | … | 0 |

MIỀN GIÁ TRỊ

Diagram

Description automatically generated

Hình ảnh được lấy từ: <https://slideplayer.com/slide/14875529/>

**DẠNG KHÔNG CHUẨN**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Tất cả bằng 0 | | 0 | 0 | … | 0 | 0 | 1 |
| CỰC TIỂU | | **Single precesion** | | | | | | | | **Double precesion** |
| S | |  | | | | | | | |  |
|  | |  | | | | | | | |  |
| Significand | |  | | | | | | | |  |
| Value | |  | | | | | | | |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | Tất cả bằng 0 | | Tất cả các bit đều là 1 |
| CỰC ĐẠI | | **Single precesion** | | **Double precesion** |
| S | |  | |  |
|  | |  | |  |
| Significand | |  | |  |
| Value | |  | |  |

**DẠNG CHUẨN**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 1 | Tất cả các bit đều là 0 |
| CỰC TIỂU | | | | | **Single precesion** | | | | | | **Double precesion** |
| S | | | | |  | | | | | |  |
|  | | | | |  | | | | | |  |
| Significand | | | | |  | | | | | |  |
| Value | | | | |  | | | | | |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 1 | 1 | 1 | | 1 | 1 | 1 | 0 | Tất cả các bit đều là 1 |
| CỰC ĐẠI | | | | | **Single precesion** | | | | | | **Double precesion** |
| S | | | | |  | | | | | |  |
|  | | | | |  | | | | | |  |
| Significand | | | | |  | | | | | |  |
| Value | | | | |  | | | | | | ( |

**ĐẢM BẢO ĐỘ CHÍNH XÁC TÍNH TOÁN BẰNG GRS**

**Nguyên nhân.**

1. Chuẩn hoá 2 số về cùng phần mũ (exponent) để thực hiện phép cộng: Dịch bit về bên phải làm cho dãy bit tràn ra ngoài và rò rĩ lượng bit này.
2. Khi ta thực hiện các phép làm tròn (Rounding & Truncate)

**Biện pháp:**

* Thêm vào cuối trong quá trình tính toán 3 bit GRS như sau.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sign | Exponent | Fraction | Guard | Round | Sticky |

* Sau khi các thao tác tính toán kết thì các extra bit này sẽ được huỷ đi (Có nghĩa là nó được sử dụng như một kết quả trung gian để phục vụ cho quá trình tính toán). Điều này giúp ta giảm thiểu phần cứng nhưng vẫn đảm bảo độ chính xác.

**Vậy 3 bit GRS là gì:**

* G: Guard bit (Chống lại sự mất mát đáng kể của việc dịch bit sang phải thoát khoải vùng lưu trữ khi thực hiện việc chuẩn hoá).
* R(Round bit) và S(Sticky bit): Đây là 2 bit cần thiết cho việc kiểm tra các đk để thực hiện việc làm tròn (rounding) mà ta đã đề cập trước đó.

|  |  |
| --- | --- |
| Guard bit | Round bit và Sticky bit |
| テキスト  低い精度で自動的に生成された説明 | テキスト が含まれている画像  自動的に生成された説明 |

Hình ảnh được lấy từ: <https://slideplayer.com/slide/14875529/>

Khảo sát sự sai số

Trong phần này ta sẽ bàn về các vấn đề sai số có thể xảy ra trên số thực:

1. Mất mát thông tin trong quá trình lưu trữ.

Biểu diễn số thực trên máy tính (Rounding and truncate error).

Ép kiểu dữ liệu (Rounding and truncate error).

Làm tròn số (Rounding and truncate error).

2. Mất mát thông tin trong quá trình tính toán.

Cộng 2 số có chênh lệch quá lớn (Phần mũ quá khác biệt) (Mất mát trung bình) (Loss of significance).

Trừ 2 số gần bằng nhau (Mất mát lớn) (Loss of significance).

Nhân có thể gây ra underflow hoặc overflow nếu phần mũ của 2 số quá lớn hoặc quá nhỏ.

Chia có thể gây ra underflow.

4. Khảo sát sự mất mát thông tin. (Đưa vào khảo sát).

Do không phải giá trị bất kì nào cũng có thể biểu diễn được trên máy tính. Cho nên với một giá trị x bất kì cần biểu diễn, ta sẽ tìm được hai giá trị a và b trên máy gần x nhất thoả điều kiện a<=x<=b và x sẽ được biểu diễn bằng a hoặc b tuỳ vào x gần giá trị nào. Ta nhận rằng giá trị ban đầu đã mất đi 1 lượng thông tin, ta tạm gọi đó là độ lỗi (error).

|  |  |
| --- | --- |
| Các loại Error | Công thức xác định |
| Rounding Error  (Absolute Error) |  |
| Relative Error |  |

**Xét số**

**Unit in the last place.**

Bất kì một giá trị x nào đó thuộc không gian A thì cũng cần phải được biểu diễn dựa trên một đơn vị nào đó, nói cách khác thì x = kUnit. Trên số thực cũng vậy, đơn vị được tính bằng khoảng cách nhỏ nhất giữa 0 và một số bất kì mà máy tính có thể biểu diễn được(dựa theo chuẩn hiện tại). đơn vị này được gọi là “unit in the last place”(ulp). Các giá trị x được biểu diễn dựa trên ulp được kí hiêu là Ulp(x), được biểu diễn như sau:

Chứng minh[8] đã chỉ ra rằng:

Khi một số trên thực tế được làm tròn đến 1 số thực gần nhất ta luôn nhận được một sai số tương đối giới hạn bởi một giá trị được gọi là hệ số epsilon machine được định nghĩa:

Để tránh những số quá nhỏ ta có thể biểu diễn lại bằng số

Ví dụ:

**Biểu diễn Rounding Error theo ULPs**

**Tính Rounding error:**

**Chuẩn hoá giá trị expected: A\*10^e**

**Tính 1 ulps:**

**Chuẩn hoá RE theo ULP**

**Ví dụ:**

**0.02 và 0.017(p=3)**

**3.14 và 3.14159(p=3)**

**99.2 và 98.8**

**12.35 và 12.4**

**0.1 và 0.0292**

**0.03 và 0.03480 (p =4)**

1. **Vấn đề Catastrophic cancellation[7]**

Xảy ra khi trừ hai số được làm tròn và hai số này gần nhau. Kết quả sẽ có thể mang lại một giá trị gần đúng rất xấu so với hiệu số của các số ban đầu.

Cho L1 = 254.5 và L2 =253.5. Nếu ta ước lượng L1=255 và L2 =253 thì

Với related error = |255-253| / L1 <2% nhưng rõ ràng kết quả ta tính ra là 2 và kết quả chính xác là 1. Điều này dẫn đến error là gấp đôi.

Chứng minh:

Gọi và với Ta có:

Do đó, related error của phép trừ hai số sấp sỉ ta có thể thấy được là: . Sai số này sẽ rất lớn và tệ hại nếu và là những số gần nhau (Có nghĩa là với ta có ).

**BẢNG TÓM TẮT VỀ MÃ HOÁ CHUẨN IEEE754**

|  |  |  |  |
| --- | --- | --- | --- |
| Single-Precision | Exponent = 8 | Fraction = 23 | Value |
| Normalized | 1…254 | Anything |  |
| Denormalized | 0 |  |  |
| Zero | 0 | 0 |  |
| Infinity | 255 | 0 |  |
| NaN | 255 |  | NaN |

|  |  |  |  |
| --- | --- | --- | --- |
| Double-Precision | Exponent = 11 | Fraction = 52 | Value |
| Normalized | 1…2046 | Anything |  |
| Denormalized | 0 |  |  |
| Zero | 0 | 0 |  |
| Infinity | 2047 | 0 |  |
| NaN | 2047 |  | NaN |

**Các quy tắc đặc biệt**

|  |  |
| --- | --- |
| **Phép tính** | **Kết quả** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Tham chiếu: <http://indico.ictp.it/event/7659/session/10/contribution/44/material/1/0.pdf>

Tài liệu:

1. <https://sites.ualberta.ca/~kbeach/comp_phys/fp_err.html>

2. <https://slideplayer.com/slide/14875529/>

3. <https://slideplayer.com/slide/14875529/>

4. <https://expressmagazine.net/posts/view/1252/cac-phep-toan-can-ban-tren-so-nhi-phan>

5. Bài giảng kiến trúc máy tính và hợp ngữ (Thầy Phạm Tuấn Sơn – ĐH Khoa Học Tự Nhiên).

6. <https://www.geeksforgeeks.org/ieee-standard-754-floating-point-numbers/>

7. <https://en.wikipedia.org/wiki/Catastrophic_cancellation>

8. <https://docs.oracle.com/cd/E19957-01/806-3568/ncg_goldberg.html>

Tool convert online:

<https://www.h-schmidt.net/FloatConverter/IEEE754.html>

**Phần viết chương trình**

Viết chương trình chuyển đổi số float.

1. Chuyển đổi từ 10 sang 2.
2. Chuyển đổi từ 2 sang 10.
3. Cho nhập vào chuẩn đang sử dụng.
4. Tính toán sai số khi chuyển đổi.
   1. Absolute error
   2. Relative error
   3. Error tính theo epsilon.
5. Hàm các phép tính trên số thực.
   1. Phép cộng
   2. Phép trừ
   3. Phép nhân
6. Hàm mô phỏng các giá trị và xuất ra bản báo cáo.

Phần sử dụng báo cáo đã duyệt để kiểm chứng các kết quả thực nghiệm.

Reference: <https://www.geeksforgeeks.org/program-for-conversion-of-32-bits-single-precision-ieee-754-floating-point-representation/>