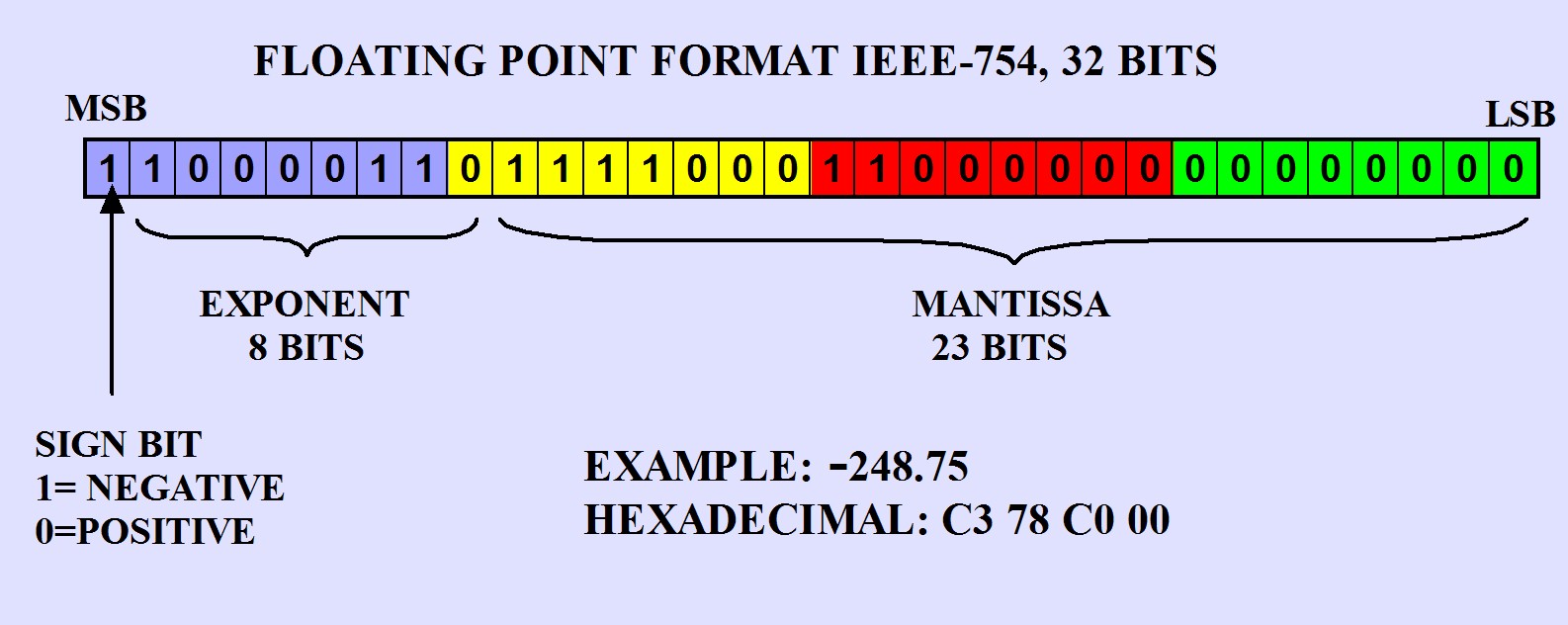
**Bài luận nghiên cứu về Floating Point**

MSSV: 21120201

Tên: Bùi Đình Bảo

1. Floating Point là gì?

Floating Point *(Dấu phẩy động ở Việt Nam hay Dấu chấm động theo các nước phương Tây)* là một thuật ngữ tin học được dùng để chỉ một hệ thống biểu diễn số mà trong đó sử dụng một chuỗi chữ số (hay bit) để biểu diễn một số hữu tỉ.



Floating Points không chỉ được ứng dụng trong việc biểu diễn số thập phân trong máy tính mà còn giúp máy tính hình thành nên tiêu chuẩn lưu trữ dữ liệu và hoạt động được lập trình sẵn.

Xuyên suốt chiều dài lịch sử phát triển của máy tính khoa học, các định dạng dấu phẩy động cũng ngày càng được cải tiến và tối ưu hóa ứng dụng vào máy tính. Cụ thể như:

* Năm 1938: nhà phát minh Konrad Zuse ở Berlin đã tạo ra "Z1", máy tính cơ lập trình được mã hóa nhị phân đầu tiên, làm việc trên các số dấu phẩy động 22 bit (phần mũ 7 bit – phần định trị 15 bit).
* Năm 1941: máy tính Z3 được hoàn thành, ứng dụng dấu phẩy động trong việc xử lý dữ liệu trong máy tính, phục vụ nhu cầu của con người.
* Năm 1962: dòng máy tính UNIVAC 1100/2200 ra đời cho phép làm việc với 2 định dạng dấu phẩy động dùng 72 bit (1 bit dấu, 11 bit phần mũ, 60 bit phần định trị).
* Năm 1980: chuẩn dấu phẩy động IEEE 754 được phát minh, sử dụng định dạng độ chính xác 32 bit (hay 16 bit và 64 bit) trở thành tiêu chuẩn của hầu hết các máy tính hiện nay, với rất nhiều những đặc điểm ưu việt. [1]

1. Floating Point trong kiến trúc máy tính nói chung

Hiệp hội IEEE đã chuẩn hóa cho việc biểu diễn số dấu phẩy động nhị phân trong máy tính bằng cách đưa ra chuẩn IEEE 754. Ngày nay hầu hết các máy tính đều tuân thủ theo chuẩn này. Một số trường hợp ngoại lệ như máy tính lớn IBM và máy vector Cray. Loại máy tính lớn IBM ngoài định dạng thập phân và nhị phân IEEE 754 còn có một định dạng riêng của IBM. Còn với máy vector Cray thì họ T90 có một phiên bản IEEE nhưng máy SV1 vẫn còn dùng định dạng dấu phẩy động của chính Cray.

Chuẩn IEEE 754 đưa ra nhiều định dạng rất gần nhau, chỉ khác nhau ở một ít chi tiết. Năm trong số những định dạng này được gọi là *định dạng cơ bản*, và hai trong chúng đặc biệt được dùng rộng rãi trong cả phần cứng máy tính và ngôn ngữ lập trình:

* Độ chính xác hơn, được gọi bằng tên là "float" trong họ ngôn ngữ lập trình C và tên là "real" hay "real\*4" trong ngôn ngữ Fortran. Đây là định dạng nhị phân chiếm 32 bit (4 byte) và phần định trị của nó có độ chính xác 24 bit (tương đương với khoảng 7 chữ số thập phân).
* Độ chính xác kép, được gọi bằng tên là "double" trong họ ngôn ngữ lập trình C và tên là "double precision" hay "real\*8" trong ngôn ngữ Fortran. Đây là định dạng nhị phân chiếm 64 bit (8 byte) và phần định trị của nó có độ chính xác 53 bit (tương đương với khoảng 16 chữ số thập phân).

Các định dạng khác là nhị phân với độ chính xác bậc bốn (128 bit), cũng như là dấu phẩy động thập phân (64 bit) và dấu phẩy động thập phân "kép" (128 bit).

Các định dạng ít thông dụng hơn:

* Định dạng độ chính xác mở rộng, mỗi số chiếm 80 bit.
* Định dạng bán chính xác cũng gọi là dấu phẩy động 16, mỗi số chiếm 16 bit.

Bất kỳ một số nguyên nào có giá trị tuyệt đối nhỏ hơn hay bằng 224 đều có thể biểu diễn một cách chính xác bằng định dạng độ chính xác đơn, và bất kỳ số nguyên nào có giá trị tuyệt đối nhỏ hơn hay bằng 253 cũng có thể biểu diễn một cách chính xác bằng định dạng độ chính xác kép.

Mặc dầu các định dạng 32 bit ("đơn") và 64 bit ("kép") hiện nay là phổ biến, nhưng chuẩn IEEE 754 cũng cho phép nhiều mức chính xác khác nhau. Lấy ví dụ, các phần cứng máy tính như họ Pentium Intel và họ 68000 Motorola thường có thêm định dạng độ chính xác mở rộng 80 bit, với phần mũ 15 bit, phần định trị 64 bit (không có bit ẩn) và 1 bit dấu. Một dự án nhằm mục đích sửa đổi chuẩn IEEE 754 đã được khởi động trong năm 2000. Dự án này đã hoàn thành và được công nhận vào tháng 6 năm 2008. Nó bao gồm các định dạng dấu phẩy động thập phân và định dạng dấu phẩy động 16 bit ("nữa"). Định dạng 16 bit nhị phân có cùng cấu trúc và quy luật như các định dạng cũ khác với 1 bit dấu, phần mũ 5 bit và 10 bit phần định trị. Định dạng này hiện đang được sử dụng trong ngôn ngữ đồ họa Cg của NVIDIA, và có mặt trong chuẩn mở EXR. [2]

1. Floating Point trong biểu diễn số học trong lập trình

Chúng ta vẫn thường được biết đến rất nhiều về số chấm động trong các bài giới thiệu tin học của chương trình học phổ thông, vậy số chấm động (Floating Point Numbers) là gì?

Số chấm động có dạng S-E-M, trong đó:

**S**: phần định trị là số **âm**, nên s là **1**

**E**: phần mũ được xác định E = **11**+127=138=**10001010**

**M**: phần định trị được xác định là **001 0010 1001 001**0 0000 0000 (23 số)

(tức là M = dãy số sau dấu chấm sau khi chuẩn hóa, còn lại ghi số 0 cho đủ 23 số)

Dãy số là:

**S** **E**              **M**

**1** **10001010** **00100101001001**000000000

**Chuyển sang hệ thập phân:**

**1** **10001010** **00100101001001**000000000

* **B1:** Coi số đầu tiên:

Số **1**: dấu **-**

Số **0**: dấu **+**

* **B2:**

Lấy dãy tiếp theo **10001010**chuyển về thập phân -> lấy số vừa chuyển về trừ 127 ra số phần mũ

**10001010 = 138**

**138**-127 = **11**

**=> Số mũ 11**

* **B3:**

Lấy dãy tiếp theo**00100101001001 đếm đủ 11 số ngăn vô dấu phẩy 00100101001.001**

* **B4:**

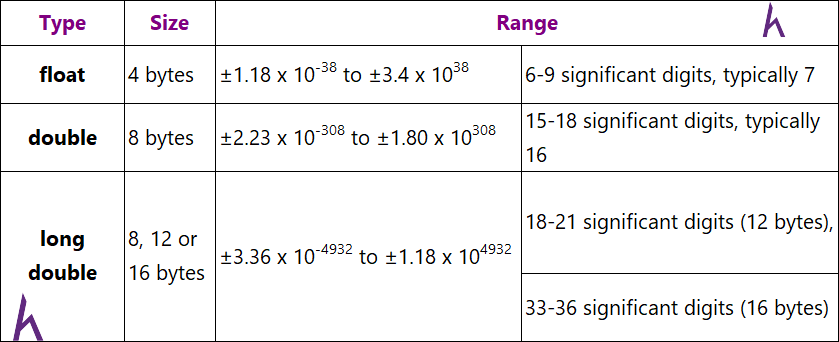
Thêm số1 (bài nào cũng thêm số 1, số 1 này không phải là số 1 ở B1) và dấu ở B1 vào dãy vừa ngăn làm ta và đổi sang thập phân

=>**-** 1**00100101001.001 = - 2345.125** [3]

Số chấm động được áp dụng vào trong các ngôn ngữ lập trình nói chung để có thể lưu trữ tối ưu nhất những con số dưới dạng nhị phân máy tính có thể hiểu được mà vẫn tối ưu được không gian bộ nhớ của thiết bị điện tử.

Ví dụ như trong C++, ngôn ngữ lập trình khởi nguyên của nhiều người thì kiểu số chấm động đại diện cho số thực (Ví dụ: 69.9696, 3.14159, 0.00001 …), dùng để lưu trữ những số rất lớn hoặc rất nhỏ. Cấu trúc lưu trữ bên trong của số thực được thiết kế theo chuẩn số chấm động (floating-point) của IEEE.

**Số chấm động không có**từ khóa **unsigned.** Có 3 kiểu số chấm động khác nhau trong C++: **float, double, long double**.



**Chú ý:** Một số môi trường lập trình đồng nhất kiểu long double với kiểu double nên kiểu này ít được sử dụng trong lập trình ứng dụng.

Cách để định nghĩa một biến số chấm động:

// Definitions of floating point numbers   
float fVarName;  
double dVarName2;  
long double ldVarName3;

**Chú ý:**

Khi bạn sử dụng một **hằng số dưới dạng một số chấm động**, quy ước số đó **phải có ít nhất 1 chữ số thập phân**, điều này giúp phân biệt số chấm động và số nguyên.

**Ví dụ:**

// Initializations of floating point numbers

**float** fVarName{4.0f};  // 4.0 means floating point (f suffix means float)

**double** dVarName2{4.0}; // 4.0 means floating point (double by default)

**long** **double** dVarName3{4.0L}; // 4.0 means floating point (L suffix means long double)

**int** nVarName4{4};            // 4 means integer

1

2

3

4

5

**Chú ý: Mặc định** một **hằng** **số thực sẽ là kiểu double**. Để có một số thực kiểu **float**, bạn cần **thêm hậu tố ‘f’**.

* **Ký hiệu khoa học (Scientific notation)**

Ký hiệu khoa học là cách xử lý những số rất lớn hoặc rất nhỏ.**Ví dụ:** chu kỳ xoay mặt trăng của Mộc Tinh là 152853.5047 s. Khi đó, bạn có thể viết bằng ký hiệu khoa học là 1.528535047 × 105 s. Hay một số khá quen thuộc với bạn như khối lượng của một electron là 9.1093822 x 10-31. Bên dưới là một số ví dụ khác:

24327 = 2.4327 x 104

7354 = 7.354 x 103

0,0078 = 7.8 x 10**–**3

0,00069 = 6.9 x 10**-**4

**Chú ý:** Số mũ sẽ là dương nếu dấu thập phân chuyển sang phải, là âm nếu dấu thập phân chuyển sang trái.

Trong C++, bạn có thể sử dụng ký hiệu khoa học để gán giá trị cho biến số chấm động. Dùng ký hiệu ‘e’ hoặc ‘E’ để thay cho 10.

**Ví dụ:**

// Initializations of floating point numbers

**double** dVarName1{69000.0};

**double** dVarName2{6.9e4};    // 6.9e4 is equal to 69000.0

**double** dVarName3{0.00069};

**double** dVarName4{6.9E-4};   // 6.9e-4 is equal to 0.00069

1

2

3

4

5

6

1. Kết luận

Rõ ràng, kỹ thuật Floating Point trong lĩnh vực toán tin học thật sự rất có giá trị trong sự phát triển của máy tính điện tử nói riêng và CNTT nói chung. Mặc dù Floating Point hiện tại vẫn có những hạn chế nhất định trong việc lưu trữ dữ liệu nhưng có thể trong tương lai gần nhất con người có thể “phá vỡ những rào cản” và khám phá những công nghệ mới thậm chí còn tốt hơn cả Floating Point! [4]

Tham khảo

|  |  |
| --- | --- |
| [1] | “Science Direct,” [Trực tuyến]. Available: https://www.sciencedirect.com/topics/engineering/floating-point-number. |
| [2] | “Geeksforgeeks,” [Trực tuyến]. Available: https://www.geeksforgeeks.org/ieee-standard-754-floating-point-numbers/. |
| [3] | “N.M.T,” [Trực tuyến]. Available: https://sites.google.com/site/nmtlxag/hoc-tap/kien-truc-may-tinh/bieu-dien-so-dau-cham-dong. |
| [4] | “Kteam,” [Trực tuyến]. Available: https://www.howkteam.vn/course/so-tu-nhien-va-so-cham-dong-trong-c-integer-floating-point/so-tu-nhien-va-so-cham-dong-trong-c-integer-floating-point-106. |