Centro Federal de Educação Tecnológica de Minas Gerais
ENGENHARIA DA COMPUTAÇÃO

Aula 02

Sistemas de Números no Computador

Ponto Flutuante

Na representação dos números em um sistema computacional, existe um modo de armazená-los em uma forma padronizada para que as operações possam ser efetuadas de maneira mais organizada, dentro da estrutura de funcionamento da máquina.

Ponto Flutuante

Na representação dos números em um sistema computacional, existe um modo de armazená-los em uma forma padronizada para que as operações possam ser efetuadas de maneira mais organizada, dentro da estrutura de funcionamento da máquina.

Uma vez que a capacidade de armazenar dados de qualquer equipamento é limitada, isso faz com que calculadoras e computadores possuam um número finito de dígitos para representar os **números**.

Ponto Flutuante

Na representação dos números em um sistema computacional, existe um modo de armazená-los em uma forma padronizada para que as operações possam ser efetuadas de maneira mais organizada, dentro da estrutura de funcionamento da máquina.

Uma vez que a capacidade de armazenar dados de qualquer equipamento é limitada, isso faz com que calculadoras e computadores possuam um número finito de dígitos para representar os **números**.

O sistema mais utilizado pelos computadores modernos é o chamado sistema de **aritmética de ponto flutuante (APF),** tanto para a representação dos números quanto para a execução das operações. **No Brasil “vírgula flutuante”.**

Ponto Flutuante

Principal Vantagem:

A principal vantagem da representação em Ponto Flutuante é devida a possibilidade de se representar em um computador tanto números muito grande quanto frações.

Ponto Flutuante

Principal Vantagem:

A principal vantagem da representação em Ponto Flutuante é devida a possibilidade de se representar em um computador tanto números muito grande quanto frações.

Principal Desvantagem:

O custo computacional.

Ponto Flutuante

Um computador ou mesmo uma calculadora representa um número real no sistema denominado **ponto flutuante**, cuja representação é dada por:

$$x = \pm 0.\, d_1 d_2 \ldots d_t \times \beta^e$$

onde:

β : é a base do sistema de numeração;

t : é o número de dígitos na mantissa;

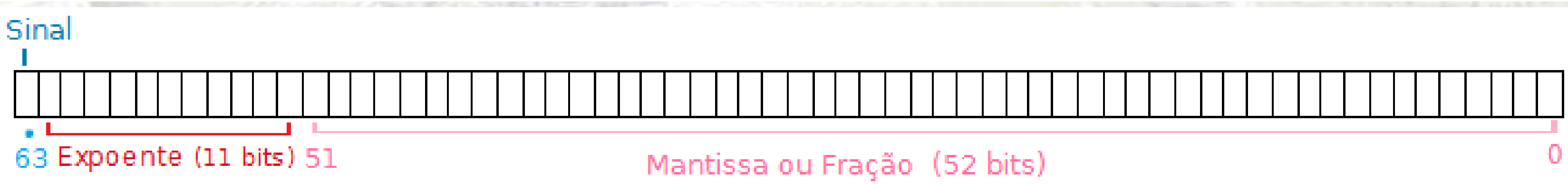
e : é o expoente, no intervalo [-m, M], em geral, m=-M.

Ponto Flutuante

IEEE Standard for Floating-Point Arithmetic (IEEE 754)

IEEE 754 - 2008	Nome usual	Tipo de dado em C++	Base b	Precisão p	Épsilon de máquina[b] $b^{-(p-1)}$
binary16	meia precisão	indisponível	2	11 (um bit implícito)	2^{-10} = 9.77e-04
binary32	precisão singular	float	2	24 (um bit implícito)	2^{-23} = 1.19e-07
binary64	precisão dupla	double	2	53 (um bit implícito)	2^{-52} = 2.22e-16
binary80	precisão estendida	_float80[4]	2	64	2^{-63} = 1.08e-19
binary128	precisão quádrupla	_float128[4]	2	113 (um bit implícito)	2^{-112} = 1.93e-34
decimal32	precisão singular decimal	_Decimal32[5]	10	7	10^{-6}
decimal64	precisão dupla decimal	_Decimal64[5]	10	16	10^{-15}
decimal128	precisão quádrupla decimal	_Decimal128[5]	10	34	10^{-33}

Precisão Dupla:



Ponto Flutuante

Exemplos: Escrever os números na notação em ponto flutuante, na base 10, considerando 5 bits na mantissa:

a) 0.55

b) –6.123

c) 0.0345

d) 6543.5

Ponto Flutuante

Exemplos: Escrever os números na notação em ponto flutuante, na base 10, considerando 5 bits na mantissa:

a) 0.55 0.55000×10^0

b) –6.123 -0.61230×10^1

c) 0.0345 0.03450×10^0

d) 6543.5 0.65435×10^4

Ponto Flutuante

Exemplos: Escrever os números na notação em ponto flutuante, na base 10, considerando 5 bits na mantissa:

a) 0.55 $\quad 0.55000 \times 10^{0}$

b) –6.123 $\quad -0.61230 \times 10^{1}$

c) 0.0345 $\quad 0.03450 \times 10^{0}$

d) 6543.5 $\quad 0.65435 \times 10^{4}$

Essa é a melhor forma de se representar um número?

Ponto Flutuante

Normalmente, a mantissa é normalizada se ela possuir o algarismo dominante nulo.

Exemplo:

$$\frac{1}{34} = 0,029411765$$

Ponto Flutuante

Normalmente, a mantissa é normalizada se ela possuir o algarismo dominante nulo.

Exemplo:

$$\frac{1}{34} = 0,029411765$$

Em Ponto Flutuante, com 4 bits na mantissa, temos:

Ponto Flutuante

Normalmente, a mantissa é normalizada se ela possuir o algarismo dominante nulo.

Exemplo:

$$\frac{1}{34} = 0,029411765$$

Em Ponto Flutuante, com 4 bits na mantissa, temos:

$$0,0294 \times 10^{0}$$

Onde pode ser observado um “zero” inútil e a perda de um bit significativo.

Ponto Flutuante

Padronização:

- O número zero pertence a qualquer sistema;
- $d_1 \neq 0$ caracteriza o sistema de números em ponto flutuante **normalizado.**
- A notação de sistema de números em ponto flutuante **normalizado,** com base β, com t dígitos significativos e com limites de expoentes m e M é:

$$F(\beta, t, m, M)$$

Ponto Flutuante

Exercícios: Escrever os números abaixo na forma em ponto flutuante normalizada F(10,3,2,2).

a) 0.55

b) –6.12

c) 0.0345

d) 6543.5

Ponto Flutuante

Exercícios: Escrever os números abaixo na forma em ponto flutuante normalizada F(10,3,2,2).

a) 0.55 0.550×10^{0}

b) –6.12 -0.612×10^{1}

c) 0.0345 0.345×10^{-1}

d) 6543.5 não pode ser representado!

Ponto Flutuante

Exercícios: Escrever os números abaixo na forma em ponto flutuante normalizada F(10,3,2,2).

a) 0.55 0.550×10^{0}

b) –6.12 -0.612×10^{1}

c) 0.0345 0.345×10^{-1}

d) 6543.5 não pode ser representado!

Neste caso, quando o expoente é menor que o valor ***m*** ocorre o que chamamos de ***underflow***. Por outro lado, quando o expoente é maior que ***M*** temos um ***overflow***.

Ponto Flutuante

Exercícios: Quantos números podem ser representados na forma em ponto flutuante normalizada F(2,3,1,2)?

Ponto Flutuante

Exercícios: Quantos números podem ser representados na forma em ponto flutuante normalizada F(2,3,1,2)?

$$x = \pm 0.\, d_1 d_2 \ldots d_t \times \beta^e$$

Solução:

2 possibilidades para o sinal;

1 possibilidade para d_1;

2 possibilidades para d_2;

2 possibilidades para d_3;

4 possibilidades para o expoente.

Ponto Flutuante

Exercícios: Quantos números podem ser representados na forma em ponto flutuante normalizada F(2,3,1,2)?

$$x = \pm 0.\, d_1 d_2 \ldots d_t \times \beta^e$$

Solução: **32 números + 0!!! Total de 33 números.**

2 possibilidades para o sinal;

1 possibilidade para d_1;

2 possibilidades para d_2;

2 possibilidades para d_3;

4 possibilidades para o expoente.

Ponto Flutuante

Conjunto Hipotético de Números em Ponto Flutuante:

Exercício: Crie um conjunto de números em **ponto flutuante normalizado** para uma máquina que armazena informações utilizando 7 bits, sendo:

1º bit para o sinal;

2º, 3º e 4º bits para o módulo do expoente;

5º, 6º e 7º bits para o módulo da mantissa.

Sendo 0 para números positivos e 1 para negativos.

Ponto Flutuante

Conjunto Hipotético de Números em Ponto Flutuante:

Solução: O menor número positivo possível será:

Ponto Flutuante

Conjunto Hipotético de Números em Ponto Flutuante:

Solução: O menor número positivo possível será:

0	1	1	1	1	0	0

Ponto Flutuante

Conjunto Hipotético de Números em Ponto Flutuante:

Solução: O menor número positivo possível será:

$$\boxed{0}\boxed{1}\boxed{1}\boxed{1}\boxed{1}\boxed{0}\boxed{0}$$

Embora uma mantissa menor seja possível (000, 001, 010, 011) a mantissa 100 é imposta pela normatização.

Ponto Flutuante

Conjunto Hipotético de Números em Ponto Flutuante:

Solução: Os números maiores são:

$$
\begin{cases}
0111\,100 = 0{,}500 \times 2^{-3} = 0{,}0625_{10} \\
0111\,101 = 0{,}625 \times 2^{-3} = 0{,}078125_{10} \\
0111\,110 = 0{,}750 \times 2^{-3} = 0{,}09375_{10} \\
0111\,111 = 0{,}875 \times 2^{-3} = 0{,}109375_{10}
\end{cases}
$$

Ponto Flutuante

Conjunto Hipotético de Números em Ponto Flutuante:

$$\begin{cases} 0110\,100 = 0,500 \times 2^{-2} = 0,125_{10} \\ 0110\,101 = 0,625 \times 2^{-2} = 0,15625_{10} \\ 0110\,110 = 0,750 \times 2^{-2} = 0,1875_{10} \\ 0110\,111 = 0,875 \times 2^{-2} = 0,21875_{10} \end{cases}$$

Ponto Flutuante

Conjunto Hipotético de Números em Ponto Flutuante:

$$\begin{cases} 0110\,100 = 0,500 \times 2^{-2} = 0,125_{10} \\ 0110\,101 = 0,625 \times 2^{-2} = 0,15625_{10} \\ 0110\,110 = 0,750 \times 2^{-2} = 0,1875_{10} \\ 0110\,111 = 0,875 \times 2^{-2} = 0,21875_{10} \end{cases}$$

Sendo o maior número a ser representador:

$$\boxed{0}\boxed{0}\boxed{1}\boxed{1}\boxed{1}\boxed{1}\boxed{1} = 7_{10}$$

Ponto Flutuante

Principais aspectos da representação Ponto Flutuante :

1) Existem um intervalo limitado de quantidades que podem ser representadas, sendo que há números positivos e negativos que não podem ser representados.

Ponto Flutuante

Principais aspectos da representação Ponto Flutuante :

1) Existem um intervalo limitado de quantidades que podem ser representadas, sendo que há números positivos e negativos que não podem ser representados.

2) Existem apenas um número finito de quantidades que podem ser representadas no intervalo, portanto, **a precisão é limitada**. Alguns números não podem ser representados, provocando **erros de quantização**.

Ponto Flutuante

Principais aspectos da representação Ponto Flutuante :

1) Existem um intervalo limitado de quantidades que podem ser representadas, sendo que há números positivos e negativos que não podem ser representados.

2) Existem apenas um número finito de quantidades que podem ser representadas no intervalo, portanto, **a precisão é limitada**. Alguns números não podem ser representados, provocando **erros de quantização**.

3) O intervalo entre os números, Δx, aumenta quando o módulo dos números cresce.

Ponto Flutuante – Padrão IEEE

A base numérica utilizada no padrão IEEE754 é a **binária**.

Ponto Flutuante – Padrão IEEE

A base numérica utilizada no padrão IEEE754 é a **binária**.

Utiliza a notação científica, não representando o bit à esquerda da vírgula decimal, sendo:

$$1,bbbbbbbb \times 2^{bbbb}$$

Ponto Flutuante – Padrão IEEE

A base numérica utilizada no padrão IEEE754 é a **binária**.

Utiliza a notação científica, não representando o bit à esquerda da vírgula decimal, sendo:

$$1,bbbbbbbb \times 2^{bbbb}$$

Exemplo:

$$1344 = 1,3125 \times 2^{10} = 1,0101 \times 2^{1010}$$

$$0,3125 = 1,25 \times 2^{-2} = 1,01 \times 2^{-10}$$

Ponto Flutuante – Padrão IEEE

Armazenamento na memória do computador:

São armazenados os valores do expoente e da mantissa separadamente, não sendo armazenado o primeiro 1 à frente da vírgula decimal.

Ponto Flutuante – Padrão IEEE

Armazenamento na memória do computador:

São armazenados os valores do expoente e da mantissa separadamente, não sendo armazenado o primeiro 1 à frente da vírgula decimal.

Os números são armazenados em precisão simples (cadeia de 32 bits) ou em precisão dupla (cadeia de 64 bits), sendo que em ambos os casos o bit mais significativo armazena o sinal (0 para positivo e 1 para negativo).

Ponto Flutuante – Padrão IEEE

Armazenamento na memória do computador:

São armazenados os valores do expoente e da mantissa separadamente, não sendo armazenado o primeiro 1 à frente da vírgula decimal.

Os números são armazenados em precisão simples (cadeia de 32 bits) ou em precisão dupla (cadeia de 64 bits), sendo que em ambos os casos o bit mais significativo armazena o sinal (0 para positivo e 1 para negativo).

Em precisão simples os próximos 8 bits armazenam o expoente e os 23 seguintes para a mantissa.

Ponto Flutuante – Padrão IEEE

O valor da mantissa é fornecido na **forma binária**.

Ao valor do expoente é acrescida a polarização (bias), que é a adição de uma valor constante para evitar o teste de sinal do expoente (que pode ser positivo ou negativo). Essa notação é conhecida como **notação em excesso.**

Ponto Flutuante – Padrão IEEE

O valor da mantissa é fornecido na **forma binária**.

Ao valor do expoente é acrescida a polarização (bias), que é a adição de uma valor constante para evitar o teste de sinal do expoente (que pode ser positivo ou negativo). Essa notação é conhecida como **notação em excesso.**

Na precisão simples, 8 bits, o valor 127 (1111111_2) é utilizado como polarização.

Ponto Flutuante – Padrão IEEE

O valor da mantissa é fornecido na **forma binária**.

Ao valor do expoente é acrescida a polarização (bias), que é a adição de uma valor constante para evitar o teste de sinal do expoente (que pode ser positivo ou negativo). Essa notação é conhecida como **notação em excesso.**

Na precisão simples, 8 bits, o valor 127 (1111111_2) é utilizado como polarização.

Exemplo: para armazenar um expoente 3 utiliza-se:

$$e = (3)_{10} = (11)_2 \text{ é armazenado como: } (1111111)_2 + (11)_2 = (10000010)_2$$

Ponto Flutuante – Padrão IEEE

Importante: existem alguns expoentes reservados!

O expoente reservado para o zero é:

$$00000000$$

0	00000000	00000000000000000000000

O expoente reservado para o $\pm\infty$ é:

$$11111111$$

1	11111111	00000000000000000000000
0	11111111	00000000000000000000000

Ponto Flutuante – Padrão IEEE

O maior expoente, em 8 bits, é:

$$11111110 = 254 - 127 = 127$$

Então o maior expoente é: +127

O menor expoente, em 8 bits, é:

$$00000001 = 1 - 127 = -126$$

Então o menor expoente é: -126

Ponto Flutuante – Padrão IEEE

O maior número positivo, em precisão simples, é:

0	11111110	11111111111111111111111

equivalente a: 1,111... x 2^{+127} = 1,7 x 10^{+38}

O menor número positivo, em precisão simples, é:

0	00000001	00000000000000000000000

equivalente a: 1,0 x 2^{-126} = 1,2 x 10^{-38}