Arquitetura e Organização de Computadores

Sistemas de Numeração: Representação em Ponto Flutuante

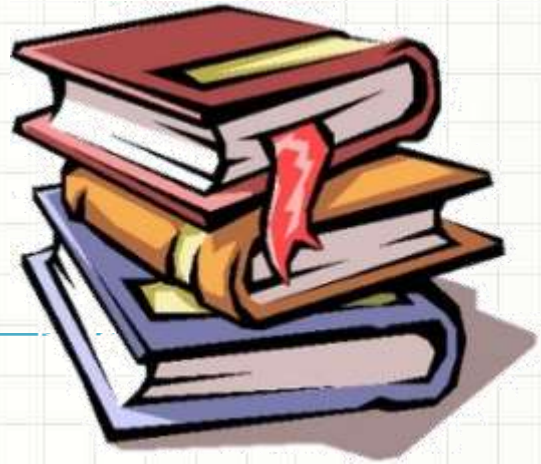
Prof. Dr. Daniel Caetano

2012 - 2

Objetivos

- Compreender o que é notação em ponto flutuante
- Compreender a representação binária de ponto flutuante segundo os padrões IEEE
- Compreender a representação do zero em ponto flutuante
- Conhecer a representação de caracteres
- **Lembrete:**
  - **Lista de Exercícios 1**!

Material de Estudo



Material	Acesso ao Material
Notas de Aula	http://www.caetano.eng.br/ (Aula 5)
Apresentação	http://www.caetano.eng.br/ (Aula 5)
Material Didático	-
Arquitetura e Organização dos Computadores	Biblioteca Virtual, páginas 289 a 292.

MOTIVAÇÃO

Motivação

- Sabemos que a memória do computador não é infinita...
- Assim, os números são guardados com um número limitado de bits
- Isso significa que números muito grandes, com muitas casas decimais, não podem ser guardados no computador

- **Será?**

O Que é Ponto Flutuante?

Notação em Ponto Flutuante

- Como guardar esse número?
- 6534000000000000000000000000000000
- Pode ser reescrito assim:

$$\mathbf{6{,}534 * 10^{+31}}$$

- O que significa **$* 10^{+31}$** ?
- Significa que tenho que mudar a vírgula de lugar para obter o número real
- No caso, preciso andar 31 casas **à direita**

Notação em Ponto Flutuante

- Como guardar esse número?
- 0,00000000000000000000000000006534
- Pode ser reescrito assim:

$$\mathbf{6,534 * 10^{-31}}$$

- O que significa **$* 10^{-31}$** ?
- Significa que tenho que mudar a vírgula de lugar para obter o número real
- No caso, preciso andar 31 casas **à esquerda**

Notação em Ponto Flutuante

- Qual o nome dessa representação?
- **$6{,}534 * 10^{+31}$**
- **$6{,}534 * 10^{-31}$**

- Notação científica ou...
- Notação em Ponto Flutuante
- Por que “ponto flutuante”?
- Por que a posição da vírgula muda (flutua) de acordo com o expoente da potência de 10

Notação em Ponto Flutuante

- É comum especificar esses números assim:
- **$6{,}534 * 10^{+31}$** → 6,534**E**31
- **$6{,}534 * 10^{-31}$** → 6,534**E-**31

- Repare que essa representação tem 3 partes

6,534**E-**31

- Quais são os nomes destas partes?

Notação em Ponto Flutuante

6,534**E**-31

- Característica
- Mantissa
- Expoente

- Qualquer número pode ser escrito assim?

Notação em Ponto Flutuante

- Vejamos alguns exemplos

Tradicional	Científica	Característica	Mantissa	Expoente
0,234	2,34**E**-1	2	34	-1
0,054334	5,4334**E**-2	5	4334	-2
1	1,0**E**0	1	0	0
10	1,0**E**1	1	0	1
100	1,0**E**2	1	0	2
1000	1,0**E**3	1	0	3
125	1,25**E**2	1	25	2
...	1,25**E**56	1	25	56
...	1,25**E**-56	1	25	-56

Ponto Flutuante em Binário

Notação em Ponto Flutuante

- Será que isso funciona com binários?
- 10110000000000000000000000000000b
- Pode ser reescrito assim:

$$\mathbf{1{,}011b * 2^{+31}}$$

- O que significa **$* 2^{+31}$** ?
- Significa que tenho que mudar a vírgula de lugar para obter o número real
- No caso, andar 31 bits **à direita**

Notação em Ponto Flutuante

- Será que isso funciona com binários?
- 0,00000000000000000000000001011b
- Pode ser reescrito assim:

$$\mathbf{1{,}011b * 2^{-31}}$$

- O que significa $\mathbf{* 2^{-31}}$?
- Significa que tenho que mudar a vírgula de lugar para obter o número real
- No caso, andar 31 bits **à esquerda**

Notação em Ponto Flutuante

- **1,011b * 2^{+31}**
- **1,011b * 2^{-31}**

- Os nomes permanecem os mesmos
- Característica
- Mantissa
- Expoente

Notação em Ponto Flutuante

- Vejamos alguns exemplos

Tradicional	Científica	Característica	Mantissa	Expoente
100b	$1,00b * 2^{2}$	1b	00b	2
101b	$1,01 * 2^{2}$	1b	01b	2
11,101b	$1,1101b * 2^{1}$	1b	1101b	1
1b	$1,0b * 2^{0}$	1b	0b	0
0,1001b	$1,001b * 2^{-1}$	1b	001b	-1
...	$1,0101b * 2^{56}$	1b	0101b	56
...	$1,0101b * 2^{-56}$	1b	0101b	-56

- O que tem de estranho aí?

Notação em Ponto Flutuante

- Vejamos alguns exemplos

Tradicional		Expoente
100b		
101b		
11,101b		
1b		
0,1001b		-1
...	1,010	1b 56
...	1,01 2^{-56}	01b 56

Se é sempre 1b, não precisamos guardar!

- O que tem de estranho aí?

Notação em Ponto Flutuante

- Assumimos que a característica é sempre 1b

Tradicional	Científica	Mantissa	Expoente
100b	$1,00b * 2^{2}$	00b	2
101b	$1,01 * 2^{2}$	01b	2
11,101b	$1,1101b * 2^{1}$	1101b	1
1b	$1,0b * 2^{0}$	0b	0
0,1001b	$1,001b * 2^{-1}$	001b	-1
...	$1,0101b * 2^{56}$	0101b	56
...	$1,0101b * 2^{-56}$	0101b	-56

Representação em Ponto Flutuante

- Como representar estes números na memória?
- Em ponto fixo, reservamos um bit para o sinal... E os demais bits para a magnitude

Bit	7 (Sinal)	6	5	4	3	2	1	0
Dígito Binário	0	1	1	0	0	0	0	1

+ ou -
Bit 7: Sinal

0 a 127
Bits 0 a 6: Magnitude

Representação em Ponto Flutuante

- O princípio é o mesmo, mas agora vamos reservar bits para várias outras coisas

Sinal	Sinal do Expoente	Expoente		Mantissa			
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

- Observe que ficaram poucos bits para cada indicação
- Expoente de -3 a +3
- Mantissa de 0000b a 1111b
- Característica -1b ou +1b

Representação em Ponto Flutuante

- Vamos representar o número 2,25 nessa notação
- Primeiro vamos converter para binário
- Parte inteira:

  2 = 10b
- Parte fracionária:

  0,25 = 0,01b
- Logo... 2,25 = 10,01b
- Em notação científica:

  $10,01b = 1,001b * 2^{1}$

Representação em Ponto Flutuante

- Bem, então 2,25 = 1,001b * 2^1
- Vamos dividir este número em partes

  Sinal: 0 (positivo) **Característica:** 1b

  Mantissa: 001b

  Sinal Expoente: 0 (positivo) **Expoente:** 1b

- Agora... Na memória:

Sinal	Sinal do Expoente	Expoente		Mantissa			
0	0	**0**	1	0	0	1	**0**
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

Representação em Ponto Flutuante

- Bem, em
- Vam
  - **Sinal:** 1b
  - **Manti**
  - **Sinal Expoen** b
- Agora... Na nem .

Nunca remova zeros à esquerda na mantissa!

Sinal	Sinal do Expoente	Expoente		Mantissa			
0	0	**0**	1	0	0	1	**0**
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

NOTAÇÃO IEEE 754/2008

Representação IEEE 754/2008

- Usar 8 bits é didático, mas muito limitado para a prática

Sinal	Sinal do Expoente	Expoente		Mantissa			
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

- IEEE define dois padrões de números de ponto flutuante:
  - **Precisão Simples (32 bits)**
  - **Precisão Dupla (64 bits)**

Representação IEEE 754/2008

- Precisão Simples (32 bits)

Sinal	Sinal do Expoente	Expoente	Mantissa
Bit 31	Bit 30	Bit 29 ~ Bit 23	Bit 22 ~ Bit0

- 1 bit para o sinal do número
- 8 bits para o expoente (incluindo sinal)
- 23 bits para a mantissa

Representação IEEE 754/2008

- Precisão Dupla (64 bits)

Sinal	Sinal do Expoente	Expoente	Mantissa
Bit 63	Bit 62	Bit 61 ~ Bit 52	Bit 51 ~ Bit0

- 1 bit para o sinal do número
- 11 bits para o expoente (incluindo sinal)
- 52 bits para a mantissa

Representação IEEE 754/2008

- Comparativo

Tipo	Bits de Sinal	Bits de Expoente	Bits de Mantissa
Simples	1	8	23
Dupla	1	11	52

- Por que aumentar mais os bits de mantissa que os de expoente?
- PRECISÃO

Representação IEEE 754/2008

- Com 11 bits de expoente, já é possível indicar 1024 casas binárias...
- É dígito que não acaba mais!

- Entretanto, não adianta ter 1 milhão de bits de expoente e ter apenas 1 bit de mantissa, pois é a mantissa que dá “detalhes” do número...

Representação IEEE 754/2008

- **Nenhum bit de expoente**:
  - Caso especial dos números em ponto fixo, o que é muito útil!
    - Podemos representar todos os números inteiros!
- **Nenhum bit de mantissa**:
  - Temos o valor 1 (característica) multiplicado pelas potências de 2 (expoente)...
  - Ou seja, podemos representar apenas potências de 2, o que é bem limitado!
    - 1, 2, 4, 8, 16, 32, 64...

Representação IEEE 754/2008

- **DETALHE MUITO IMPORTANTE**
- Nas representações anteriores, Bit de Sinal:
  - 0 é positivo
  - 1 é negativo
- Na notação IEEE, isso vale para o sinal do número...
- Mas no sinal do **expoente**, o bit de sinal é **invertido**, ou seja:
  - 0 é negativo
  - 1 é positivo

Representação IEEE 754/2008

- **DETALHE IMPORTANTE**
- Até
  - 0 é p
  - 1 é n
- Na notação [illegible] al do número
- No sinal do expoente, o bit de sinal é invertido, ou seja:
  - 0 é negativo
  - 1 é positivo

Ué?! Por quê? Para complicar a vida?

Representação do Zero

Representação do Zero

- Voltemos ao nosso caso com apenas 8 bits

Sinal	Sinal do Expoente	Expoente		Mantissa			
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

- Zero seria $0{,}0b * 2^0$

  Sinal: 0 (positivo) **Característica:** 0b

  Mantissa: 0b

  Sinal Expoente: 1 (positivo) **Expoente:** 0b

- **Como indicar característica 0b !?!?!**

Representação do Zero

- IEEE determinou que se precisarmos indicar característica 0b...
- ... Usaremos o expoente **-0b** !
- Assim, o número ZERO pode ser escrito como

Sinal	Sinal do Expoente	Expoente		Mantissa			
0	0	0	0	0	0	0	0
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

- Note que o uso de bit de sinal invertido no expoente (0 é negativo) permite que o valor 0 seja representado como “tudo 0”!

Representação do Zero

- Expoente ≠ -0: **notação normalizada**
  - Característica é 1b
- Expoente = -0: **notação não-normalizada**
  - Característica é 0b

Contas em Ponto Flutuante

Cálculos em Ponto Flutuante

- Uma série de transformações são necessárias...
- Basicamente, os elementos precisam ter o mesmo expoente para serem somados
- Complicado!
- Por isso a maioria dos computadores tem uma...
- FPU – Floating Point Unit

Representação de Caracteres

Representação de Caracteres

- Vimos como representar números...
- Mas como representar letras?
- Problema antigo: surgiu com a computação
- Tabela ASCII
  - American Standard for Computer Information Interchange
- Cada um dos códigos visuais de caracteres são mapeados para um número

Representação de Caracteres

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Source: www.LookupTables.com

Representação de Caracteres

- Esta tabela define os caracteres de 0 a 127
- Os caracteres de 128 a 255 são “extras”
- Cada país implementou a sua extensão, para os seus acentos, chamada “codepage”

- Isso criou muita confusão e, então, criaram os padrões mundiais UNICODE
- Os tipos comuns são UTF-8, UTF-16 e UTF-32

Representação de Caracteres

- UTF: Unicode Transformation Format
  - UTF-8: 256 caracteres
  - UTF-16: 65536 caracteres
  - UTF-32: 4 bilhões de caracteres
- UTF-8 é compatível com ASCII

  (Apenas os 128 primeiros caracteres do ASCII)
- UTF-16 é compatível com UTF-8
- UTF-32 é compatível com UTF-16

EXERCÍCIO

Exercício

1. Represente o número **273,5234** segundo padrão IEEE de 32 bits

2. Escreva a palavra **Abacaxi** como o computador a vê, isto é, usando os códigos ASCII dos caracteres. Use a notação hexadecimal

Conclusões

Resumo

- É possível representar números muito grandes e muito pequenos com o uso de ponto flutuante
- O formato padrão de representar ponto flutuante no computador é o estabelecido pela IEEE 754/2008.
- Os caracteres também são representados como números ... E existem várias codificações possíveis

- **TAREFA**
  - **Lista 1!**

Próxima Aula

- Ok... Mas como é o lugar onde estes números ficam?
  - Como eles são escritos?
  - Como eles são acessados?

PERGUNTAS?

Bom Descanso a Todos!