Circuitos Lógicos

Aula 5

cruz@gta.ufrj.br http://gta.ufrj.br/~cruz

Poli – Dept. de Eng. Eletrônica e de Computação
Universidade Federal do Rio de Janeiro

Na última aula

- Transformando um problema numa expressão
  - Mintermos
  - Maxtermos
- Implementando expressões booleanas com portas lógicas

- Simplificação de expressões booleanas
  - Comutatividade
  - Associatividade
  - Teorema de De Morgan
  - Equivalência XOR
  - ....

Hoje

- Codificação
- Representando números inteiros com 0’s e 1’s
  - Conversão binário – decimal
  - Conversão binário – hexadecimal
  - Representação de números negativos
    - Sinal-magnitude
    - Complemento a 2
  - Código de Gray
- Representação de caracteres

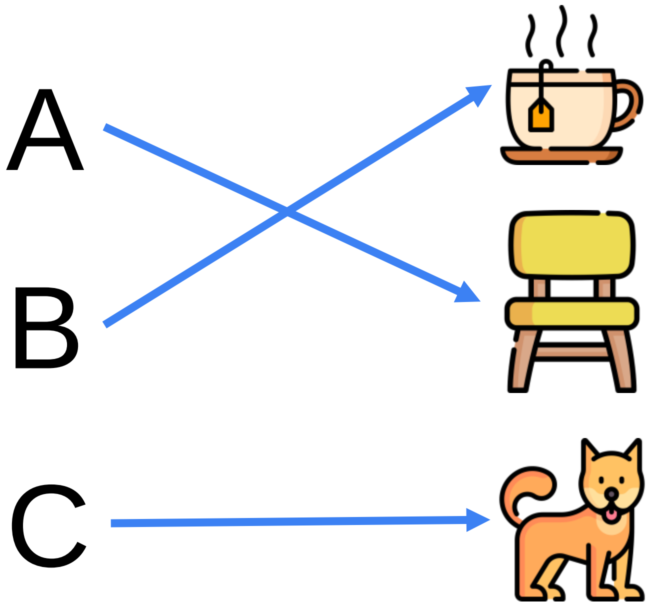
Assuntos administrativos

- Prática 2 hoje!
- Prova 1 dia 16 ou 18 de maio?
  - Revisão na aula anterior
    - “Pergunte qualquer coisa”
- Trabalho 1
  - Entrega 30 de maio (mais de 1 mês para fazer)
- Prova 2
  - 6 de julho
- Prova final
  - 13 de julho
- Prova de 2ª chamada
  - 20 de julho
- Entrega do trabalho prático 2
  - 18 de julho

Codificação

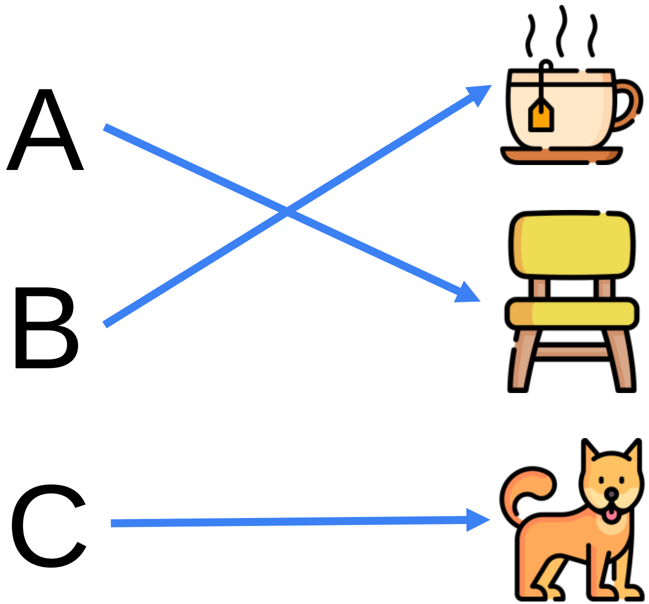
Codificar

- Sejam F e D alfabetos fonte e destino
- Uma codificação é uma associação de cada símbolo de F a cada símbolo de D
  - Função bijetora



Codificar

- Alfabetos podem ser quaisquer coisas imagináveis
  - Letras
  - Algarismos
  - Números
  - Palavras
  - Sons faláveis
- Codificação também
  - Escrita
    - Letras – sons faláveis
  - Numeração
    - Quantidades – números
  - Vocabulário
    - Palavras – ideias

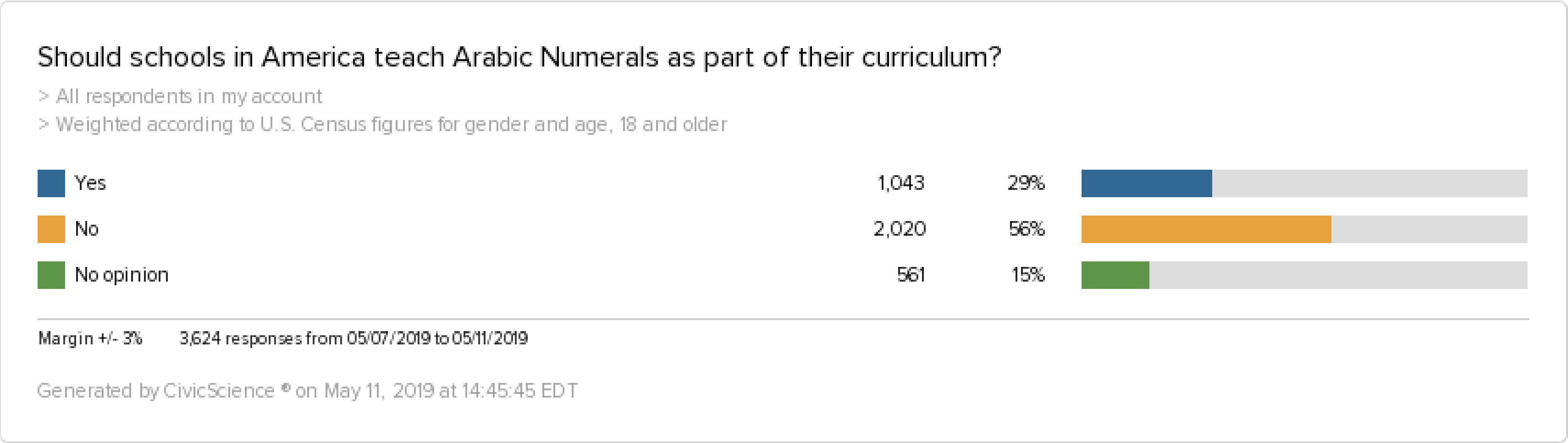


Representação em binário

- Sequência de bits representa um símbolo de um alfabeto
    - Alfabeto pode ser:
        - Números inteiros
        - Algarismos
        - Letras
        - Cores
        - Estados de uma variável
        - ....

Representação de números naturais

Pequena pesquisa



“As escolas dos EUA deveriam ensinar números arábicos como parte de seu currículo?”

Numerais arábicos

- Possui dez algarismos
  - 0,1,2,3,4,5,6,7,8,9
    - Significam quantidades básicas

- É capaz de representar o zero
  - 0

- É posicional
  - Dependendo da posição que o algarismo ocupa no número, ele tem um valor diferente
    - Unidade, dezena, centena, unidade de milhar....

Aprendendo a contar

- Alfabeto que representa quantidades
- Número de algarismos define **base**
- Contamos até o maior algarismo
  - Contar +1 “estoura”
  - Adicionamos +1 na casa à esquerda
  - Reiniciamos a contagem

Dezena	Unidade
0	0
0	1
0	2
0	3
0	4
0	5
0	6
0	7
0	8
0	9
1	0
1	1

Dezena	Unidade
1	2
1	3
1	4
1	5
1	6
1	7
1	8
1	9
2	0
2	1
2	2
2	3

Aprendendo a contar

- Alfabeto que representa quantidades
- Número de algarismos define **base**
- Contamos até o maior algarismo
  - Contar +1 “estoura”
  - Adicionamos +1 na casa à esquerda
  - Reiniciamos a contagem

- Algarismos à direita são **menos significativos**
- Algarismos à esquerda são **mais significativos**

Dezena	Unidade
0	0
0	1
0	2
0	3
0	4
0	5
0	6
0	7
0	8
0	9
1	0
1	1

Dezena	Unidade
1	2
1	3
1	4
1	5
1	6
1	7
1	8
1	9
2	0
2	1
2	2
2	3

Sistema posicional

- Cada **casa** representa uma potência da base
  - Significa um **multiplicador**

	-	...	Unidade de Milhar	Centena	Dezena	Unidade
Multiplicador	10....0	...	1000	100	10	1
Multiplicador em potência de 10	10^n	...	10^3	10^2	10^1	10^0

Sistema posicional

- Cada **casa** representa uma potência da base
  - Significa um **multiplicador**

	-	...	Unidade de Milhar	Centena	Dezena	Unidade
Multiplicador	10....0	...	1000	100	10	1
Multiplicador em potência de 10	10^n	...	10^3	10^2	10^1	10^0

Se serve pra 10, serve pra 2!

Contando em binário

- Cada **casa** representa uma potência da base
  - Significa um **multiplicador**

	6ª posição	5ª posição	4ª posição	3ª posição	2ª posição	1ª posição
Multiplicador	10....0	...	8	4	2	1
Multiplicador em potência de 10	2^n	...	2^3	2^2	2^1	2^0

Dígito na posição k é multiplicado por 2^{k-1}

Contando em binário

2^2	2^1	2^0	Binário	Decimal
0	0	0	0	0
0	0	1	1	1
0	1	0	10	2
0	1	1	11	3
1	0	0	100	4
1	0	1	101	5
1	1	0	110	6
1	1	1	111	7

Contando em binário

O valor máximo alcançado por k bits é 2^k-1

Valores maiores precisam de mais bits

2^2	2^1	2^0	Binário	Decimal
0	0	0	0	0
0	0	1	1	1
0	1	0	10	2
0	1	1	11	3
1	0	0	100	4
1	0	1	101	5
1	1	0	110	6
1	1	1	111	7

Convertendo binário para decimal

- Realizamos uma adição dos valores 1, multiplicados pelo multiplicador relativo à casa
- Exemplo: conversão do binário 1001011

$$1001011 = 1 \cdot 2^6 + 0 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$$

Então,

$$1001011 = 1 \cdot 64 + 0 \cdot 32 + 0 \cdot 16 + 1 \cdot 8 + 0 \cdot 4 + 1 \cdot 2 + 1 \cdot 1$$

$$1001011 = 75$$

Potência de 2	Decimal
2^0	1
2^1	2
2^2	4
2^3	8
2^4	16
2^5	32
2^6	64

Convertendo decimal para binário

- Método do quociente
  - Dividir o número por 2 sucessivas vezes
    - Resto será 0 ou 1
    - Resto da divisão n será o n-ésimo dígito binário
  - Quando o resultado for zero, temos a conversão

Potência de 2	Decimal
2^0	1
2^1	2
2^2	4
2^3	8
2^4	16
2^5	32
2^6	64

Convertendo decimal para binário

Exemplo: converter 75

$75 \div 2 = 37$, resto 1

$37 \div 2 = 18$, resto 1

$18 \div 2 = 9$, resto 0

$9 \div 2 = 4$, resto 1

$4 \div 2 = 2$, resto 0

$2 \div 2 = 1$, resto 0

$1 \div 2 = 0$, resto 1

$75 = 1001011$

Potência de 2	Decimal
2^0	1
2^1	2
2^2	4
2^3	8
2^4	16
2^5	32
2^6	64

Representação *Binary-Coded Decimal* (BCD)

Binary-Coded Decimal

- Cada dígito do decimal é codificado em 4 bits
  - Dígitos seguem conversão tradicional

Decimal	Representação binária
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Binary-Coded Decimal

Exemplo: converter 75

$7 = 0111$

$5 = 0101$

$75 = 0111\ 0101$

Decimal	Representação binária
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Representação em hexadecimal

Hexadecimal

- Cada 4 bits codificam um algarismo
  - 4 bits podem assumir 16 valores
    - Necessários 16 algarismos

Decimal	Binário	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Hexadecimal

- Conversão entre hexadecimal e binário
  - De 4 em 4 bits
  - Fácil
- Conversão entre hexadecimal e decimal
  - Vale o método do resto
    - Dividir por 16, guardar o resto

Decimal	Binário	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Convertendo decimal para hexadecimal

Exemplo: converter 75

$75 \div 16 = 4$, resto 11 (em hexadecimal, B)

$4 \div 16 = 0$, resto 4

$75 = 4B$H

(convenciona-se colocar H após representação hexadecimal)

n	$\times 16$
0	0
1	16
2	32
3	48
4	64
5	80
6	96

Representação de números inteiros

Representando números negativos

- Sinal-magnitude (bit de sinal)
- Complemento a/de/para 2 (diferentes literaturas usam a, de, ou para)

Sinal-magnitude

- Magnitude é representada da mesma forma que na representação normal
- Um bit adicional representa se o sinal é negativo ou positivo
- Deve ter um número fixo de bits
    - Bit de sinal têm posição fixa

Exemplo com 3 bits

Exemplo: representar -2 com 3 bits

$2 = 010$

$-2 = 110$

Código	Valor Decimal	Valor sinal-magnitude
000	0	0
001	1	1
010	2	2
011	3	3
100	4	0
101	5	-1
110	6	-2
111	7	-3

Problemas com sinal-magnitude

- Zero tem duas representações possíveis
  - Uma negativa (?) e outra positiva (?)
- Operações de soma não são triviais
  - Número negativo deve ser detectado e então subtraído

Complemento para 2

- Números positivos são representados normalmente
- Números negativos são o número positivo de mesma magnitude com duas operações
    - Inversão de cada bit
    - Adição de 1

Representação garante unicidade do 0

Representação garante soma trivial

Complemento para 2

Exemplo: converter o número 75, com 8 bits

$75 = 01001011$
$\overline{75} = 10110100$
$\overline{75} + 1 = 10110101$
$-75 = 10110101$

Código	Valor Decimal	Valor complemento para 2
000	0	0
001	1	1
010	2	2
011	3	3
100	4	-4
101	5	-3
110	6	-2
111	7	-1

Código de Gray

Contando em binário

- Contagem binária possui trocas abruptas de valores
  - A cada “estouro” de dígito significativo, todos menos significativos mudam
- No cenário com comunicação paralela, existe algum atraso entre diferentes bits
- E se só um bit mudasse por vez?

Decimal	Binário
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

Código de Gray

- Contagem binária possui trocas abruptas de valores
  - A cada “estouro” de dígito significativo, todos menos significativos mudam

Decimal	Binário	Gray
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Representação de caracteres

ASCII

American Standard Code for Information Interchange - ASCII

- Representação 7 bits
  - Letras
  - Algarismos
  - Pontuação
  - Controle
- Representado numa tabela
  - Colunas são os 4 bits menos significativos
    - Normalmente representados em hexadecimal
  - Linhas são os 3 bits mais significativos
    - Normalmente representados em decimal ou hexadecimal (não vai fazer diferença)

Tabela ASCII

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0x	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1x	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2x	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3x	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4x	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5x	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6x	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7x	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

Representação de Enumeráveis e Mensuráveis

Enumeráveis e mensuráveis

- Enumerar
  - Contar
  - Atribuir alguma ordem
- Medir
  - Estimar um tamanho

Enumeráveis e mensuráveis

- Se posso estabelecer uma ordem, posso atribuir um número
  - Posso representar esse número como binário

- Se posso estimar um tamanho, posso representar em binário
  - Atributos podem ser representados

Novas possibilidades

- Posso ordenar vários tipos de *coisas*
  - Cores
  - Países
  - Produtos de uma loja
  - ....
- Posso medir vários tipos de grandezas
  - Tamanho
  - Temperatura
  - Velocidade
  - .....
- Posso representar *coisas* em binário e estabelecer relações lógicas entre elas!
  - Cor se transforma se tem sombra
  - Altura se deforma dependendo do ângulo de visão
  - ......

Créditos

Os ícones desta apresentação foram feitos por Freepic e retirados de www.flaticon.com.

GTA / UFRJ
GRUPO DE TELEINFORMÁTICA E AUTOMAÇÃO

www.gta.ufrj.br