



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

# A unidade da informação

Notas de aula II – do BIT ao UTF

# Introdução à Computação usando Python

## A unidade da informação

---



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### O que vamos estudar nesta Aula:

- Modelo simplificado do computador
- Definição conceitual do BIT: Dígito Binário.
- Definição de Memória computacional.
- Representação prática do bit em circuitos digitais.
- Tipos de dados memorizados e suas características computacionais.
- Relação de crescimento das variáveis e proposições lógicas e suas representações em circuitos digitais.
- O processo de memorização computacional.
- Modelagem das variáveis e proposições lógicas: Tabela Verdade.
- Unidades de medida de armazenamento de dados.
- Armazenamento de dados sobre variáveis e proposições lógicas.

*Prof. Alessandro Bertolani Oliveira*

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### O que vamos estudar nesta Aula:

- Sistema numérico Binário.
- Sistema numérico Binário versus Tabela verdade.
- Sistema Numérico Binário x Decimal: Notação em base numérica.
- Sistema Numérico Binário x Decimal: Modelagem do BIT por circuitos digitais.
- Transformação Binário x Decimal:
- Mudança de base numérica.
- Transformação Decimal x Binário: Mudança de base numérica.
- Mudança de base numérica em circuitos digitais.
- Escala de unidade da informação: Múltiplos padronizados do bit.
- Padrão: ASCII.
- Padrão: ASCII Estendido.
- Padrão: BYTE.

# Introdução à Computação usando Python

## A unidade da informação

---



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### O que vamos estudar nesta Aula:

- Aplicações e exemplos dos múltiplos padronizados do BIT.
  - Padrão: UTC
  - Padrão: UTC – 8 – 16 – 32.
- Aplicações e exemplos dos múltiplos padronizados UTC.
  -

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPIRITO SANTO

### O modelo simplificado do computador:



- Valores escolhidos pelo usuário do programa.
- Dados do problema.
- Memorização: Variável.
- Tipos: Inteiro / Real / Caractere / Lógica
- Alocação de memória

- Atividades executadas pelo computador.
- Algoritmo: Sequência de Passos (Instruções): Ações + Decisões = Objetivo.
- Processamento: Cálculos sobre as Variáveis.
- Processo de Seleção e Repetição de Dados: Proposições Lógicas.

- Exibir: Objetivos do problema.
- Mostrar ao usuário na tela as soluções encontradas do problema.
- Memorizar esses **Dados processados** pelo programa em **Informação**.

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### Definição de Memorização Computacional:

É o processo de armazenamento das variáveis (Dados de entrada) / informações (Dados de saída) de um problema computacional em modo:

- **Temporário** – Volátil (RAM) e CACHE
- **Permanente** – Não Volátil (HD / SSD)



Memória volátil – DDR5



Memória não volátil – SSD

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### Características dos dados memorizados:

**Tipos de dados: valores** que são armazenados computacionalmente:

- **Inteiro** ( $\in \mathbb{Z}$ ): Alunos / Matrícula / Horas / etc...
- **Real** ( $\in \mathbb{R}$ ): Salário / Preço / Delta / etc...
- **Caractere** (Símbolo): Nome / Rua / CPF/ etc...
- **Lógica** (0 ou 1): Resposta / Lâmpada / Interruptor / etc...



# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### FOMULÁRIO I

	<b>UNIVERSIDADE VILA VELHA</b> <b>INTRODUÇÃO À CIÊNCIA DA COMPUTAÇÃO</b> <b>DESCRIÇÃO DE PROFICIÊNCIA ACADÊMICA</b>	<b>FOTO:</b>
<b>Dados do Aluno</b>		
<b>2 - NOME COMPLETO:</b>		<b>3 - FORMAÇÃO - NÍVEL MÉDIO (ano):</b>
Email:		CPF:
Endereço:	Nº	Bairro:
Município:	UF:	CEP:
Telefone:		
Renda Individual (R\$):		
Renda familiar per capita (R\$):	Renda Familiar Total (R\$):	Quantidade de indivíduos:
Possui casa própria? SIM <input type="checkbox"/> NÃO <input type="checkbox"/>		
<b>Dados de Proficiência</b>		
Já estudou Lógica de Programação (Lógica Estruturada) ? SIM <input type="checkbox"/> NÃO <input type="checkbox"/>		
Sabe programar em alguma linguagem? SIM <input type="checkbox"/> Qual(is)? (Exemplo: Python, Java, c#, outros...) NÃO <input type="checkbox"/>		

**Tipos de dados: valores em um Formulário de pesquisa**

*Prof. Alessandro Bertolani Oliveira*



# Introdução à Computação usando Python

## A unidade da informação

---



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### Dúvidas sobre o processo de Memorização:

1. Qual desses quatro (4) tipos de variáveis (dados) ocupa o “menor” espaço de memória ?
2. Como armazenar as variáveis do tipo: Inteiro / Real / Caractere / Lógica na memória do computador ?
3. Qual a unidade de medida da memória computacional ?
4. Como os Dados: valores extremamente altos são processados no ambiente de memorização computacional ? Por exemplo: Endereço IP de Site na Internet (IP: *Internet Protocol*).

# Introdução à Computação usando Python

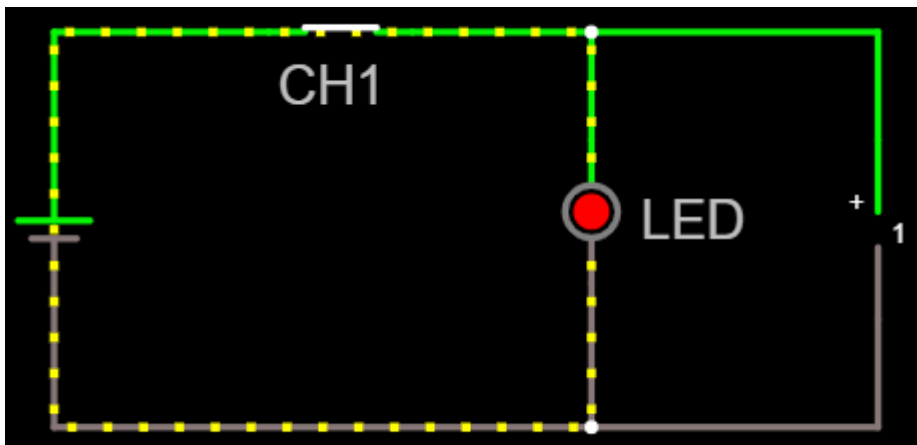
## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### Armazenamento de valores lógicos:

- O armazenamento de uma **variável lógica** ou uma **proposição lógica** é dado através do armazenamento de um **Dígito Binário** – BIT (*Blnary digiT*).
- Um bit: 0 ou 1 é o menor valor que pode ser armazenado na memória do computador. Sua representação em circuito:



CHAVE 1	LED
0	0
1	1

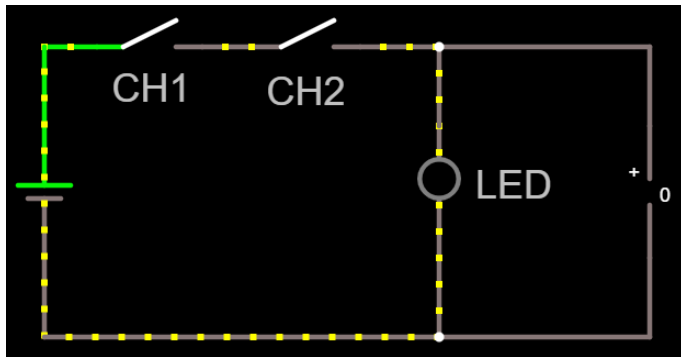
# Introdução à Computação usando Python

## A unidade da informação

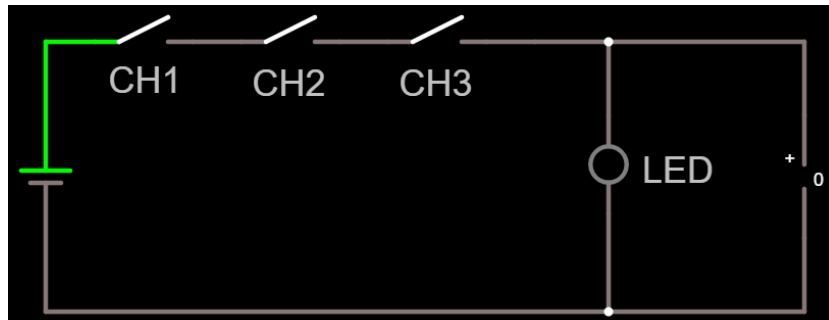


Número de variáveis lógicas:

**LED = CH1 E CH2**



**LED = CH1 E CH2 E CH3**



CHAVE 1	CHAVE 2	LED
0	0	0
0	1	0
1	0	0
1	1	1

CHAVE 1	CHAVE 2	CHAVE 3	LED
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

# Introdução à Computação usando Python

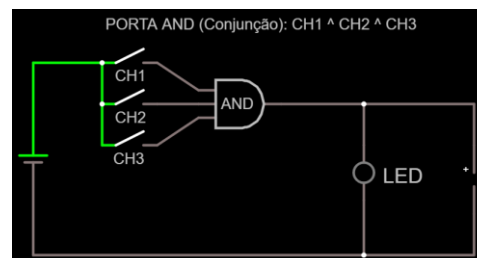
## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### CONJUNÇÃO (^): LED = CH1 E CH2 E CH3

CHAVE 1	CHAVE 2	CHAVE 3	LED
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1



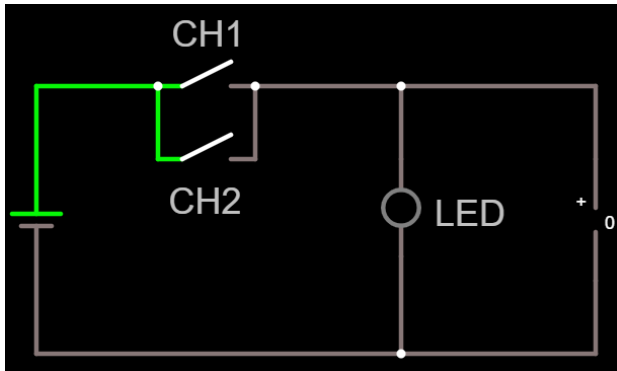
# Introdução à Computação usando Python

## A unidade da informação

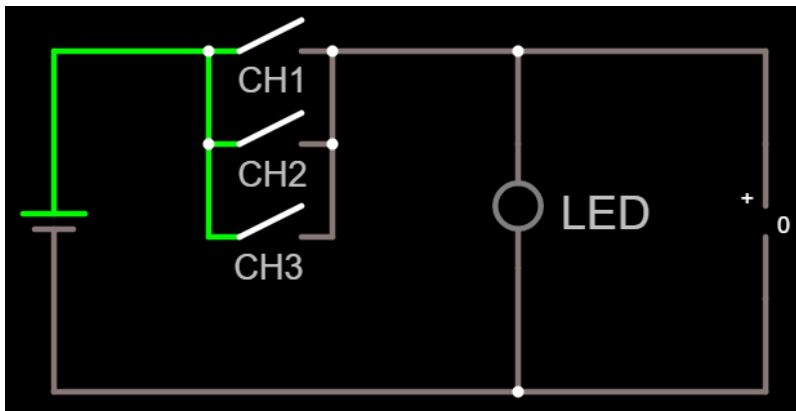


Número de variáveis lógicas:

**LED = CH1 OU CH2**



**LED = CH1 OU CH2 OU CH3**



LINK DO SITE: Circuitos lógicos

CHAVE 1	CHAVE 2	LED
0	0	0
0	1	1
1	0	1
1	1	1

CHAVE 1	CHAVE 2	CHAVE 3	LED
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

# Introdução à Computação usando Python

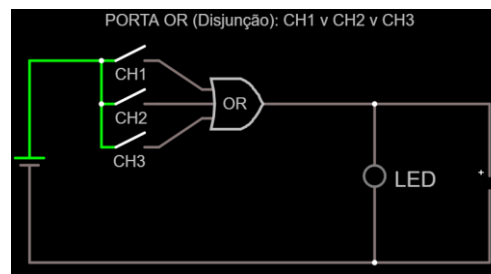
## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

**DISJUNÇÃO (V): LED = CH1 V CH2 V CH3**

CHAVE 1	CHAVE 2	CHAVE 3	LED
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1



LINK DO SITE: Circuitos lógicos

*Prof. Alessandro Bertolani Oliveira*

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### Número de variáveis lógicas:

*Número de Linhas =  $2^n$ , onde  $n$  = Número de Variáveis Lógicas ou Proposições Simples.*

$n$	1	2	3	4	5	6	7	8	9	10	...
$2^n$	2	4	8	16	32	64	128	256	512	1024	...

A modelagem e combinação de todos os resultados possíveis das variáveis lógicas damos o nome de **Tabela Verdade**.

CHAVE 1	LED
0	0
1	1

Tabela Verdade de uma (1) Variável Lógica – CHAVE 1

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### Sistema Numérico Binário:

George Boole

Matemático



George Boole foi um matemático, filósofo britânico, criador da álgebra booleana, fundamental para o desenvolvimento da computação moderna. [Wikipédia](#)

**Nascimento:** 2 de novembro de 1815, [Lincoln, Reino Unido](#)

**Falecimento:** 8 de dezembro de 1864, [Ballintemple, Cork, Irlanda](#)

**Cônjuge:** [Mary Everest Boole](#) (de 1855 a 1864)

**Filhas:** [Ethel Lilian Voynich](#), [Alicia Boole Stott](#), [Lucy Everest Boole](#), [Mary Boole Hinton](#), [Margaret Taylor](#)

**Nacionalidade:** Inglês, Britânico



# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPIRITO SANTO

### Sistema Numérico Binário:

BIT 0	BASE 10
0	0
1	1

Uma (1) Variável Lógica

BIT 1	BIT 0	BASE 10
0	0	0
0	1	1
1	0	2
1	1	3

Duas (2) Variáveis Lógicas

BIT 2	BIT 1	BIT 0	BASE 10
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Três (3) Variáveis Lógicas

*Prof. Alessandro Bertolani Oliveira*

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPIRITO SANTO

### Sistema Numérico Binário:

BIT 3	BIT 2	BIT 1	BIT 0	BASE 10
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

Quatro (4) Variáveis Lógicas

.....assim sucessivamente.

*Prof. Alessandro Bertolani Oliveira*

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

Sistema Numérico Binário x Decimal: Notação por base.

$$234_{10} = 2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0$$

$$234_{10} = 200 + 30 + 4 = 234$$

$2^2$	$2^1$	$2^0$	BASE 2	BASE 10
0	0	0	$0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 0 \times 4 + 0 \times 2 + 1 \times 1 = 0+0+0$	0
0	0	1	$0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 0 \times 4 + 0 \times 2 + 1 \times 1 = 0+0+1$	1
0	1	0	$0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 0 \times 4 + 1 \times 2 + 0 \times 1 = 0+2+0$	2
0	1	1	$0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 0 \times 4 + 1 \times 2 + 1 \times 1 = 0+2+1$	3
1	0	0	$1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 = 1 \times 4 + 0 \times 2 + 0 \times 1 = 4+0+0$	4
1	0	1	$1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 1 \times 4 + 0 \times 2 + 1 \times 1 = 4+0+1$	5
1	1	0	$1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 1 \times 4 + 1 \times 2 + 0 \times 1 = 4+2+0$	6
1	1	1	$1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 1 \times 4 + 1 \times 2 + 1 \times 1 = 4+2+1$	7

# Introdução à Computação usando Python

## A unidade da informação

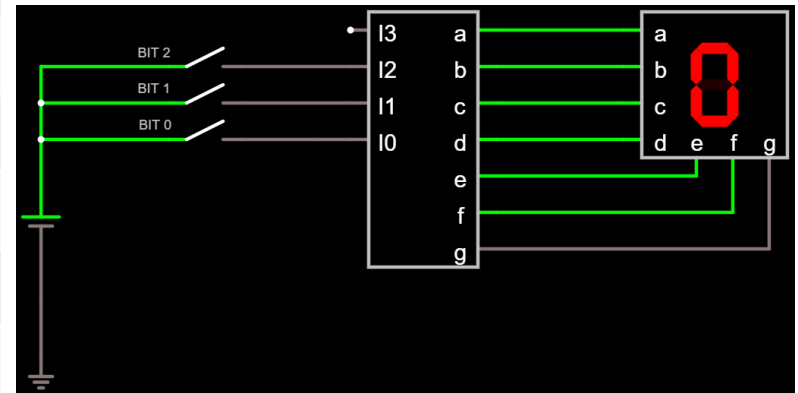


UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### Sistema Numérico Binário x Decimal:

BIT 2 = 4	BIT 1 = 2	BIT 0 = 1	BASE 10
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Três (3) Variáveis Lógicas



Display de 7 LEDS

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPIRITO SANTO

Transformação Binário x Decimal: Mudança de base numérica.

$2^2$	$2^1$	$2^0$	BASE 2	BASE 10
0	0	0	$0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 = 0 \times 4 + 0 \times 2 + 0 \times 1 = 0+0+0$	0
0	0	1	$0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 0 \times 4 + 0 \times 2 + 1 \times 1 = 0+0+1$	1
0	1	0	$0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 0 \times 4 + 1 \times 2 + 0 \times 1 = 0+2+0$	2
0	1	1	$0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 0 \times 4 + 1 \times 2 + 1 \times 1 = 0+2+1$	3
1	0	0	$1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 = 1 \times 4 + 0 \times 2 + 0 \times 1 = 4+0+0$	4
1	0	1	$1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 1 \times 4 + 0 \times 2 + 1 \times 1 = 4+0+1$	5
1	1	0	$1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 1 \times 4 + 1 \times 2 + 0 \times 1 = 4+2+0$	6
1	1	1	$1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 1 \times 4 + 1 \times 2 + 1 \times 1 = 4+2+1$	7

Como fazer a mudança de Base 2 para Base 10 ?

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### Transformação Binário x Decimal: Mudança de base numérica.

Como fazer a mudança de Base 2 para Base 10 ?

Basta somar os BITS que estão Ligados (1).

BIT 2 = 4	BIT 1 = 2	BIT 0 = 1	BASE 10
1	0	1	?
4	0	1	5

$$101_2 = 5_{10}$$

BIT 2 = 4	BIT 1 = 2	BIT 0 = 1	BASE 10
0	1	1	?
0	2	1	3

$$011_2 = 3_{10}$$

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### Transformação Binário x Decimal: Mudança de base numérica.

Como fazer a mudança de Base 2 para Base 10 ?

Basta somar os **BITs** que estão **Ligados (1)**.

BIT 3 = 8	BIT 2 = 4	BIT 1 = 2	BIT 0 = 1	BASE 10
1	0	0	1	?
8	0	0	1	9

$$1001_2 = 9_{10}$$

BIT 3 = 8	BIT 2 = 4	BIT 1 = 2	BIT 0 = 1	BASE 10
1	1	1	0	?
8	4	2	0	14

$$1110_2 = 14_{10}$$

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

Transformação Binário x Decimal: Mudança de base numérica.

Como fazer a mudança de Base 2 para Base 10 ?

**Somar os BIT's que estão Ligados (1).**

1	0	0	0	1	1	BASE 10
32	16	8	4	2	1	$32 + 2 + 1 = 35$

$$100011_2 = 35_{10}$$

Tente você agora ?  $101000_2 = ?_{10}$



# Introdução à Computação usando Python

## A unidade da informação

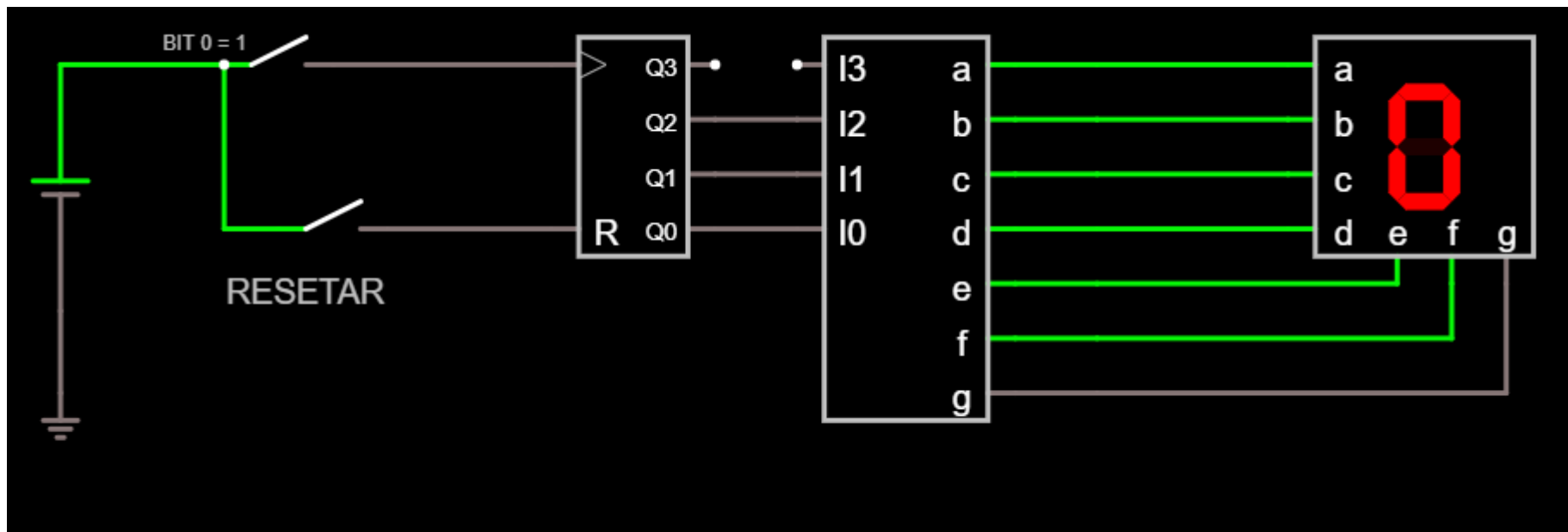


UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

Transformação Decimal x Binário: Mudança de base numérica.

**Agora ao contrário:** Como fazer a mudança de Base 10 para Base 2 ?

**Somar os BITS que estão Ligados (1).**



# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

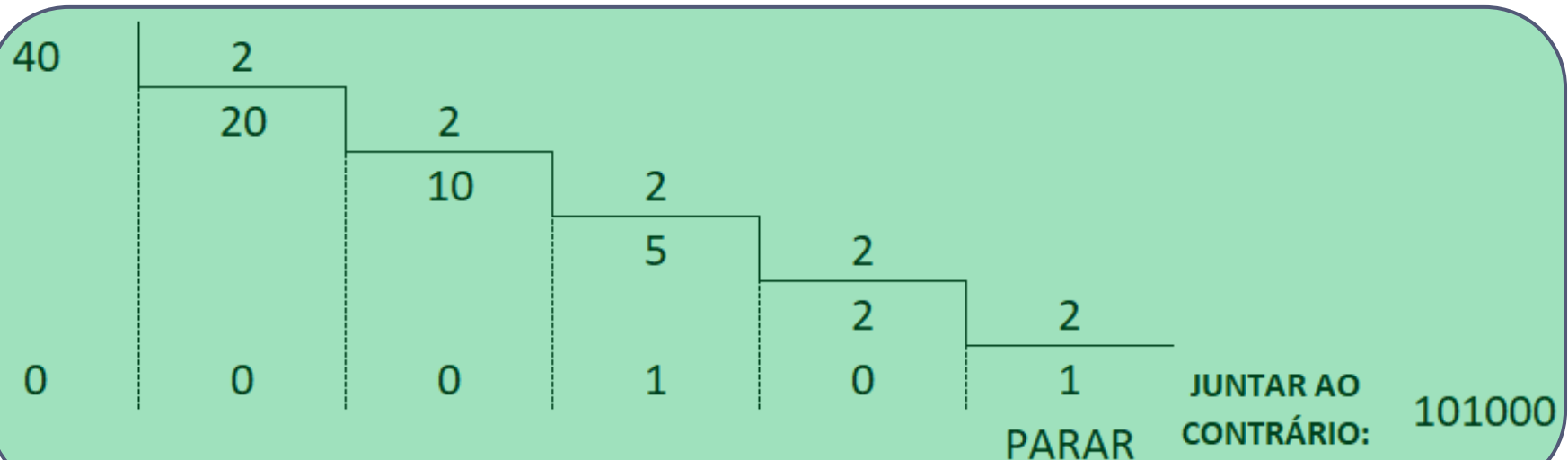
**Agora ao contrário:** Como fazer a mudança de **Base 10 para Base 2** ?

**Faça a sequência algorítmica a seguir:**

**PASSO 1:** Dividir o número sucessivas vezes por 2.

**PASSO 2:** Parar a divisão em 1.

**PASSO 3:** Juntar os 0's e 1's da Direita para Esquerda.



# Introdução à Computação usando Python

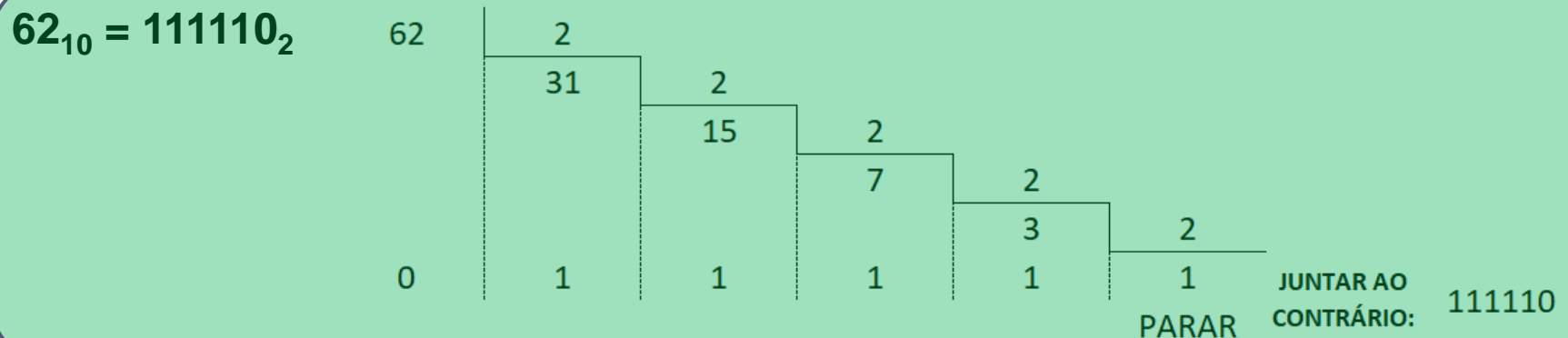
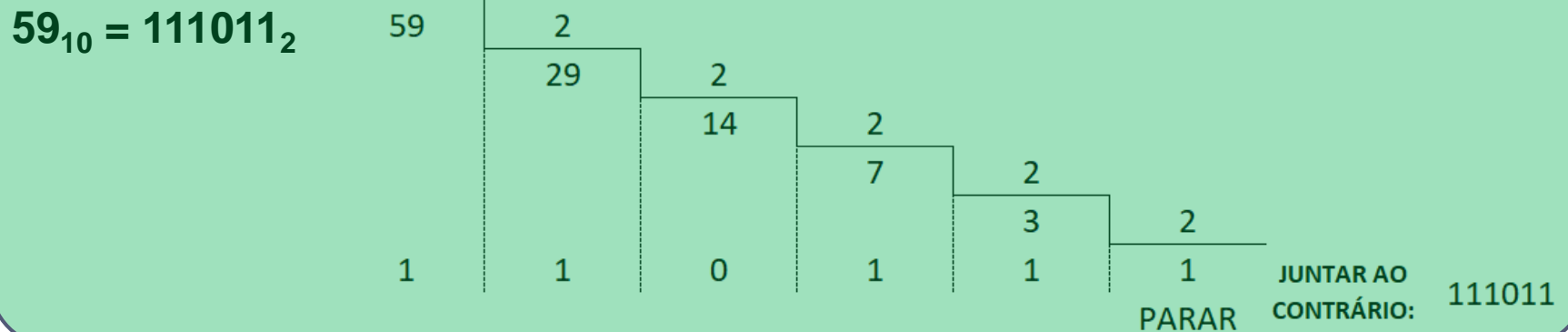
## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

**Agora ao contrário:** Como fazer a mudança de **Base 10 para Base 2** ?

Outros exemplos:



# Introdução à Computação usando Python

## A unidade da informação

---



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

**Escala de unidade da informação: Múltiplos padronizados do bit.**

➤ **PADRÃO DE 7 bits:**

**ASCII** (do inglês *American Standard Code for Information Interchange*): "Código Padrão Americano para o Intercâmbio de Informação") — geralmente pronunciado [áski] — é um código binário (cadeias de bits: 0s e 1s) que codifica um conjunto de **128 sinais**:

- **95 sinais gráficos** (letras do alfabeto latino, sinais de pontuação e sinais matemáticos) e
  - **33 sinais de controle**,
- utilizando portanto apenas **7 bits para representar todos os seus símbolos**.

# Introdução à Computação usando Python

## A unidade da informação



### Escala de unidade da informação: Múltiplos padronizados do bit.

ASCII (1977/1986)

	_0	_1	_2	_3	_4	_5	_6	_7	_8	_9	_A	_B	_C	_D	_E	_F
0_0	NUL 0000	SOH 0001	STX 0002	ETX 0003	EOT 0004	ENQ 0005	ACK 0006	BEL 0007	BS 0008	HT 0009	LF 000A	VT 000B	FF 000C	CR 000D	SO 000E	SI 000F
1_16	DLE 0010	DC1 0011	DC2 0012	DC3 0013	DC4 0014	NAK 0015	SYN 0016	ETB 0017	CAN 0018	EM 0019	SUB 001A	ESC 001B	FS 001C	GS 001D	RS 001E	US 001F
2_32	SP 0020	! 0021	" 0022	# 0023	\$ 0024	% 0025	& 0026	' 0027	( 0028	) 0029	* 002A	+ 002B	, 002C	- 002D	. 002E	/ 002F
3_48	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	: 003A	; 003B	< 003C	= 003D	> 003E	? 003F
4_64	@ 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
5_80	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[ 005B	\ 005C	] 005D	^ 005E	_ 005F
6_96	` 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
7_112	p 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	 007C	} 007D	~ 007E	DEL 007F

☐ Letter
 ☐ Number
 ☐ Punctuation
 ☐ Symbol
 ☐ Other
 ☐ undefined
 ☐ Changed from 1963 version

**ASCII - PADRÃO DE 7 BIT's:**

# Introdução à Computação usando Python

## A unidade da informação



## Escala de unidade da informação: Múltiplos padronizados do bit.

Bin	Oct	Dec	Hex	Sinal	Bin	Oct	Dec	Hex	Sinal	Bin	Oct	Dec	Hex	Sinal	Bin	Oct	Dec	Hex	Abrev	Notação com circunflexo	Código escape	Nome
0010 0000	040	32	20	(espaço)	0100 0000	100	64	40	@	0110 0000	140	96	60	`	0000 0000	000	00	00	NUL	^@	\0	Nulo (inglês Null)
0010 0001	041	33	21	!	0100 0001	101	65	41	A	0110 0001	141	97	61	a	0000 0001	001	01	01	SOH	^A		Início de cabeçalho (inglês Start of Header)
0010 0010	042	34	22	"	0100 0010	102	66	42	B	0110 0010	142	98	62	b	0000 0010	002	02	02	STX	^B		Início de texto (inglês Start of Text)
0010 0011	043	35	23	#	0100 0011	103	67	43	C	0110 0011	143	99	63	c	0000 0011	003	03	03	ETX	^C		Fim de texto (inglês End of Text)
0010 0100	044	36	24	\$	0100 0100	104	68	44	D	0110 0100	144	100	64	d	0000 0100	004	04	04	EOT	^D		Fim de transmissão (inglês End of Transmission)
0010 0101	045	37	25	%	0100 0101	105	69	45	E	0110 0101	145	101	65	e	0000 0101	005	05	05	ENQ	^E		Consulta; inquirição (inglês Enquiry)
0010 0110	046	38	26	&	0100 0110	106	70	46	F	0110 0110	146	102	66	f	0000 0110	006	06	06	ACK	^F		Confirmação (inglês Acknowledge)
0010 0111	047	39	27	'	0100 0111	107	71	47	G	0110 0111	147	103	67	g	0000 0111	007	07	07	BEL	^G	\a	Campainha; sinal sonoro (inglês Bell)
0010 1000	050	40	28	(	0100 1000	110	72	48	H	0110 1000	150	104	68	h	0000 1000	010	08	08	BS	^H	\b	Espaço atrás; retorno de 1 caractere (inglês Back-space)
0010 1001	051	41	29	)	0100 1001	111	73	49	I	0110 1001	151	105	69	i	0000 1001	011	09	09	HT	^I	\t	Tabulação horizontal (inglês Horizontal Tabulation)
0010 1010	052	42	2A	*	0100 1010	112	74	4A	J	0110 1010	152	106	6A	j	0000 1010	012	10	0A	LF	^J	\n	Alimentação de linha; mudança de linha; nova linha (inglês Line Feed)
0010 1011	053	43	2B	+	0100 1011	113	75	4B	K	0110 1011	153	107	6B	k	0000 1011	013	11	0B	VT	^K	\v	Tabulação vertical (inglês Vertical Tabulation)
0010 1100	054	44	2C	,	0100 1100	114	76	4C	L	0110 1100	154	108	6C	l	0000 1100	014	12	0C	FF	^L	\f	Alimentação de formulário (inglês Form Feed)
0010 1101	055	45	2D	-	0100 1101	115	77	4D	M	0110 1101	155	109	6D	m	0000 1101	015	13	0D	CR	^M	\r	Retorno do carro; retorno ao início da linha (inglês Carriage Return)
0010 1110	056	46	2E	.	0100 1110	116	78	4E	N	0110 1110	156	110	6E	n	0000 1110	016	14	0E	SO	^N		Mover para fora; deslocamento para fora (inglês Shift Out)
0010 1111	057	47	2F	/	0100 1111	117	79	4F	O	0110 1111	157	111	6F	o	0000 1111	017	15	0F	SI	^O		Mover para dentro; deslocamento para dentro (inglês Shift In)
0011 0000	060	48	30	0	0101 0000	120	80	50	P	0111 0000	160	112	70	p	0001 0000	020	16	10	DLE	^P		escape do linque de dados; escape de conexão (inglês Data-Link Escape)
0011 0001	061	49	31	1	0101 0001	121	81	51	Q	0111 0001	161	113	71	q	0001 0001	021	17	11	DC1	^Q		Controle de dispositivo 1 (inglês Device Control 1)
0011 0010	062	50	32	2	0101 0010	122	82	52	R	0111 0010	162	114	72	r	0001 0010	022	18	12	DC2	^R		Controle de dispositivo 2 (inglês Device Control 2)
0011 0011	063	51	33	3	0101 0011	123	83	53	S	0111 0011	163	115	73	s	0001 0011	023	19	13	DC3	^S		Controle de dispositivo 3 (inglês Device Control 3)
0011 0100	064	52	34	4	0101 0100	124	84	54	T	0111 0100	164	116	74	t	0001 0100	024	20	14	DC4	^T		Controle de dispositivo 4 (inglês Device Control 4)
0011 0101	065	53	35	5	0101 0101	125	85	55	U	0111 0101	165	117	75	u	0001 0101	025	21	15	NAK	^U		Confirmação negativa (inglês Negative-Acknowledge)
0011 0110	066	54	36	6	0101 0110	126	86	56	V	0111 0110	166	118	76	v	0001 0110	026	22	16	SYN	^V		Estado ocioso síncrono; espera síncrona (inglês Synchronous Idle)
0011 0111	067	55	37	7	0101 0111	127	87	57	W	0111 0111	167	119	77	w	0001 0111	027	23	17	ETB	^W		Bloco de fim de transmissão (inglês End of Transmission Block)
0011 1000	070	56	38	8	0101 1000	130	88	58	X	0111 1000	170	120	78	x	0001 1000	030	24	18	CAN	^X		Cancelar (inglês Cancel)
0011 1001	071	57	39	9	0101 1001	131	89	59	Y	0111 1001	171	121	79	y	0001 1001	031	25	19	EM	^Y		Fim de mídia; fim do meio (inglês End of Medium)
0011 1010	072	58	3A	:	0101 1010	132	90	5A	Z	0111 1010	172	122	7A	z	0001 1010	032	26	1A	SUB	^Z		Substituir (inglês Substitute)
0011 1011	073	59	3B	;	0101 1011	133	91	5B	[	0111 1011	173	123	7B	{	0001 1011	033	27	1B	ESC	^[		Escapar (inglês Escape)
0011 1100	074	60	3C	<	0101 1100	134	92	5C	\	0111 1100	174	124	7C		0001 1100	034	28	1C	FS	^\		Separador de arquivos (inglês File Separator)
0011 1101	075	61	3D	=	0101 1101	135	93	5D	]	0111 1101	175	125	7D	}	0001 1101	035	29	1D	GS	^]	\e	Separador de grupos (inglês Group Separator)
0011 1110	076	62	3E	>	0101 1110	136	94	5E	^	0111 1110	176	126	7E	~	0001 1110	036	30	1E	RS	^^		Separador de registros (inglês Record Separator)
0011 1111	077	63	3F	?	0101 1111	137	95	5F	_						0001 1111	037	31	1F	US	^_		Separador de unidades (inglês Unit Separator)
															0111 1111	177	127	7F	DEL	^?		Deletar (inglês Delete)

ASCII - PADRÃO DE 7 BIT's:

# Introdução à Computação usando Python

## A unidade da informação

---



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

**Escala de unidade da informação: Múltiplos padronizados do bit.**

➤ **PADRÃO DE 7 bits:**

**1963 - O Código Padrão Americano para Intercâmbio de Informações**

(ASCII) foi desenvolvido sob os cuidados de um comitê da *American Standards Association* (ASA).

O ASA tornou-se o Instituto de Padrões dos Estados Unidos da América.

Por fim; o *American National Standards Institute* (PADRÃO: ANSI).

O ASCII foi usado pela primeira vez comercialmente como um código do dispositivo eletrônico: “tele impressora: FAX” de sete bits para a rede TWX (*TeletypeWriter eXchange*) da *American Telephone & Telegraph* (AT & T).

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### Escala de unidade da informação: Múltiplos padronizados do bit.

- **1963**: ASCII - PADRÃO DE 7 bits.
- **~1968**: *Extended* ASCII - PADRÃO DE 8 bits.
- **1965**: BYTE - PADRÃO DE 8 bits. (IBM)
- **1991**: UNICODE: *Universal Coded Character Set* - PADRÃO DE 8 bits.
  - **2009**: UTF-8: Unicode Transformation Format – 8-bytes.
  - UTF-16: Unicode Transformation Format – 16-bytes.
  - UTF-32: Unicode Transformation Format – 32-bytes.
  - **2019** (Março): Unicode 12.0: **137.993 caracteres** + 150 scripts +  
símbolos múltiplos + conjunto de emoji.



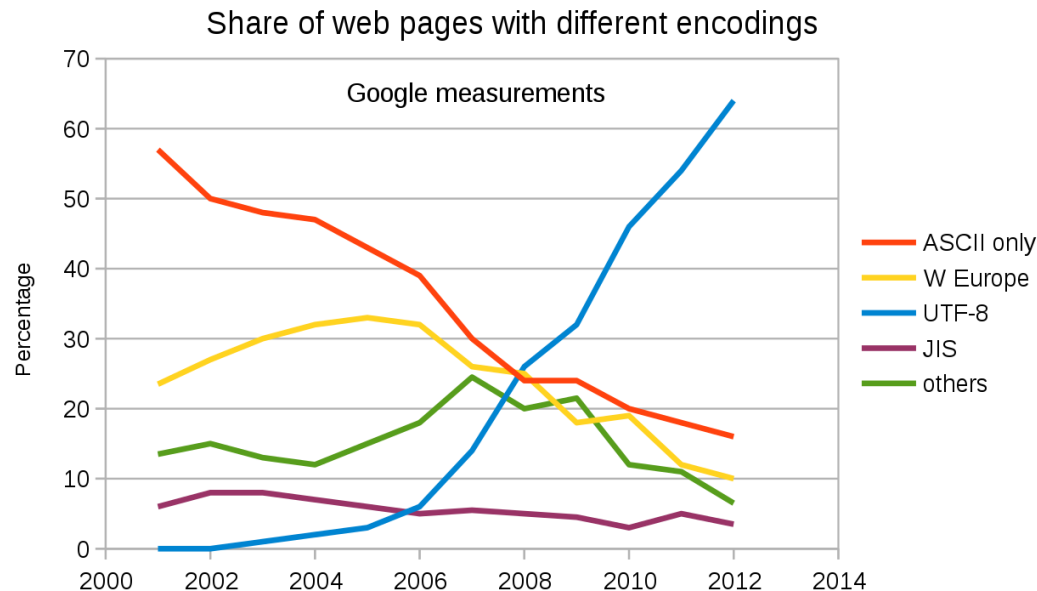
# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### Escala de unidade da informação: Múltiplos padronizados do bit.



2018: UTF-8: ["Specifying the document's character encoding"](#), *HTML 5.2*, [World Wide Web Consortium](#), 14 December 2017, retrieved 2018-06-03

<https://www.w3.org/TR/html5/index.html#contents>

**ECMAScript® 2020:** <https://tc39.github.io/ecma262/#sec-intro>

*Prof. Alessandro Bertolani Oliveira*

# Introdução à Computação usando Python

## A unidade da informação

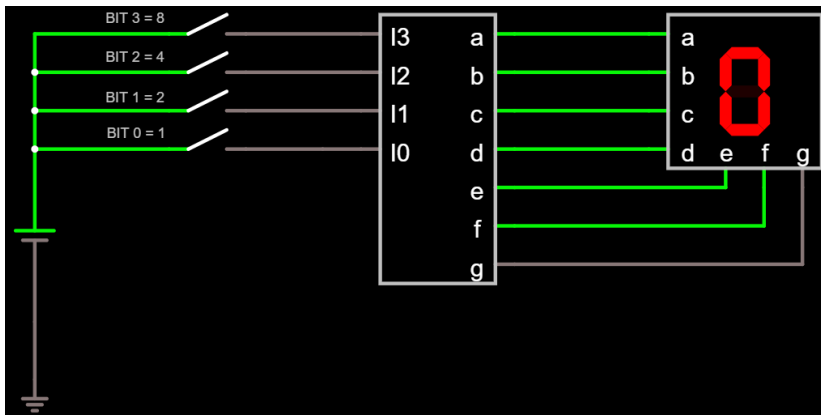


UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

**Escala de unidade da informação: Múltiplos padronizados do bit.**

### ➤ **BYTE (B): PADRÃO DE 8 Bits:**

O termo byte foi cunhado por Werner Buchholz em junho de 1956, durante a fase inicial do Projeto: computador IBM Stretch, que tinha endereçado as instruções bit e *Variable Field Length* (VFL) com um **tamanho (variável)** de byte codificado na instrução.



**PADRÃO DE 8 BIT's = 1 BYTE**



# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPIRITO SANTO

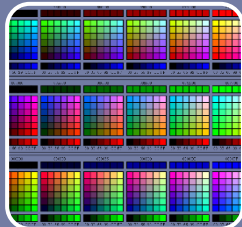
Como representar um número “muito grande” utilizando pouco espaço

de memória? **RESPOSTA: Sistema Numérico Hexadecimal (Base 16)**



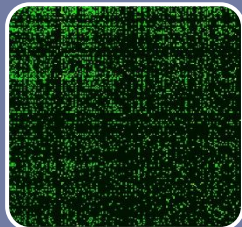
### SISTEMA NUMÉRICO DECIMAL

- Base 10: 0, 1, 2, ...9 (10 Símbolos)
- Nível de APP, GUI e Sistemas.
- Desenvolvimento de Software.



### SISTEMA NUMÉRICO HEXADECIMAL

- Base 16: 0, 1, 2, ...9, A, B, C, D, E, F (16 Símbolos)
- Sistema Operacional (SO).
- Endereço de Memória (BYTE = 8 bits): RAM, HD, Cache, Registradores.



### SISTEMA NUMÉRICO BINÁRIO

- Base 2: 0 ou 1 (2 Símbolos)
- Nível de Hardware.
- BIT (BInary digiT)

**SISTEMA NUMÉRICO COMPUTACIONAL**

# Introdução à Computação usando Python

## A unidade da informação

---



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

**Escala de unidade da informação: Múltiplos padronizados do bit.**

**BYTE (Conjunto de 8 bits):** São usados no endereçamento de memória  
(**Hexadecimal**) aplicado aos elementos de:

- **WEB:** Domínio de Site: URL (*Uniform Resource Locator*): IP Address.
- **HARDWARE:** MAC ADDRESS de uma máquina.
- **Linguagem de Programação:** Variáveis, Constantes e Proposições.
- **Linguagem de Programação:** Estrutura de Dados.
- **Padronização de Cores:** Triplete Hexadecimal (WEB COLOR).
- Entre outros....

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### Escala de unidade da informação: Múltiplos padronizados do bit.

**Tripleto hexadecimal** ou **web colors** é um número de seis dígitos formado por **três bytes** em hexadecimal. É utilizado em documentos HTML, CSS e em outras aplicações em computação.

Os bytes representam as porções das cores vermelho, verde e azul (**RGB**). Cada byte usa a faixa de 00h a FFh (notação hexadecimal) ou de 0 a 255 em notação decimal. O tripleto hexadecimal é formado pela concatenação dos **três bytes em hexadecimal**:

**Byte 1 – RED:** valor de vermelho.

**Byte 2 – GREEN:** valor de verde.

**Byte 3 – BLUE:** valor de azul.

Número de cores que podem ser representadas pelo **sistema tripleto (3 BYTES)**:

$$256 \times 256 \times 256 = 16.777.216 \text{ CORES}$$

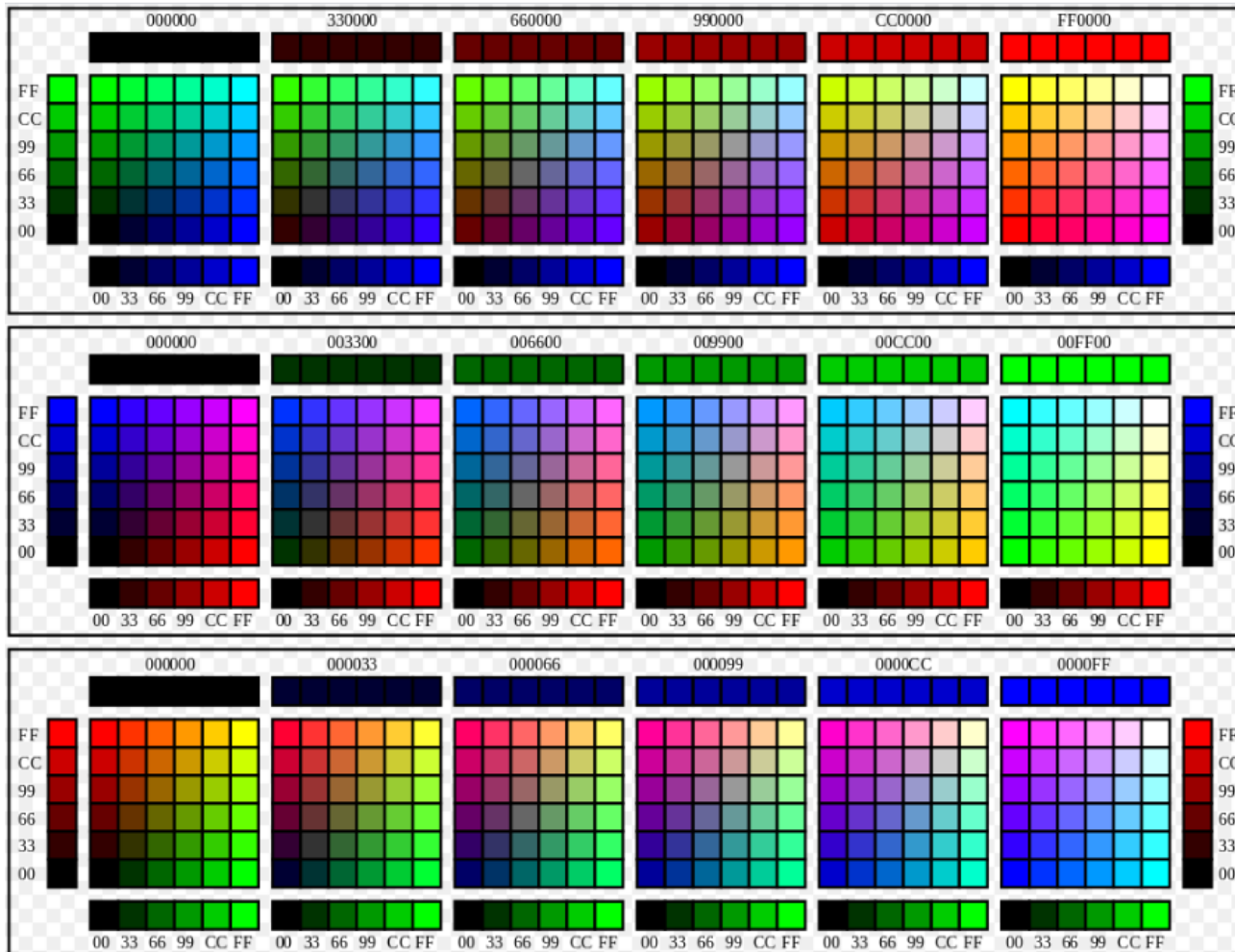
# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPIRITO SANTO

**Escala de unidade da informação: Múltiplos padronizados do bit.**



Tripleto Hexadecimal ou Web Colors.

*Prof. Alessandro Bertolani Oliveira*

# Introdução à Computação usando Python

## A unidade da informação

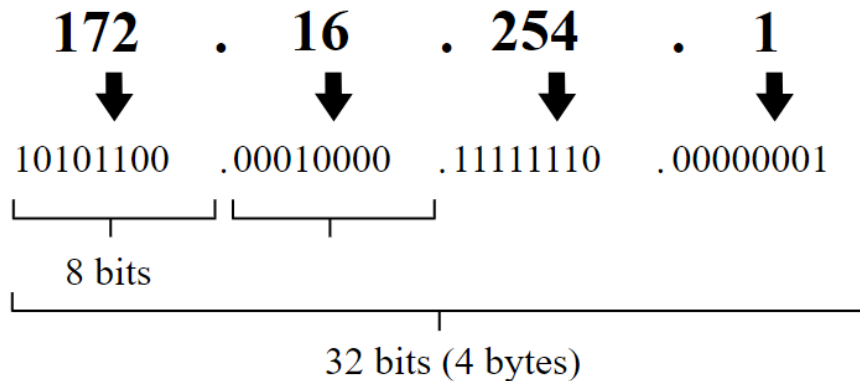


UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

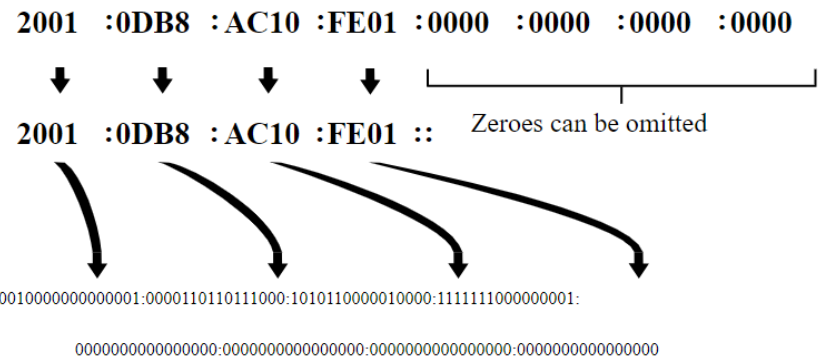
**Escala de unidade da informação: Múltiplos padronizados do bit.**

**BYTE (Conjunto de 8 bits): Na WEB, com o endereçamento de domínio de uma URL por exemplo em IPv4 e IPv6:**

IPv4 address in dotted-decimal notation



An IPv6 address (in hexadecimal)



Exemplo de armazenamento em memória de endereços de máquina: Domínio WEB (URL).



# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

**Escala de unidade da informação: Múltiplos padronizados do bit.**

**BYTE (Conjunto de 8 bits): Em Hardware, com o endereçamento de máquinas:**



Exemplo de armazenamento em memória de endereços de máquina: *IP Address*.

# Introdução à Computação usando Python

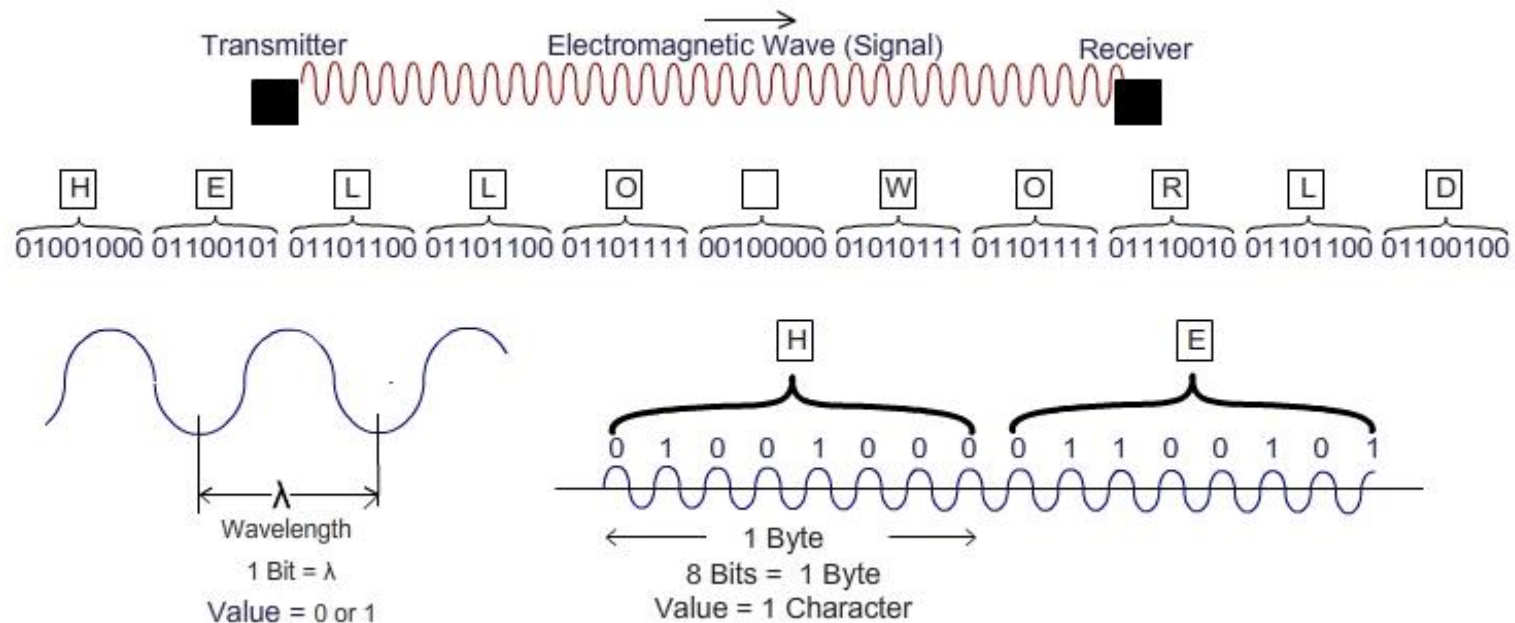
## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

**Escala de unidade da informação: Múltiplos padronizados do bit.**

**BYTE (Conjunto de 8 bits):** Na Linguagem de Programação, o formato de um (1) BYTE (Rodar o Programa: Table Extended ASCII) é:



Um (1) **Caractere** ASCII 'H':  $01001000_2 = 72_{10}$  é armazenado em exatamente um (1) Byte.

# Introdução à Computação usando Python

## A unidade da informação

---



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

**Escala de unidade da informação: Múltiplos padronizados do bit.**

**BYTE (Conjunto de 8 bits):** Na Linguagem de Programação, o formato das variáveis são:

- **Caractere:** Símbolos individuais do padrão ASCII estendido.

TAMANHO EM Bytes: 1 byte.

- **\*\*\* Inteiro:** Números inteiros entre -2147483648 e 2147483647.

TAMANHO EM Bytes: 4 Bytes.

- **\*\*\* Real:** reais entre (aproximadamente)  $10^{-38}$  e  $10^{38}$ .

TAMANHO EM Bytes: 4 Bytes.

**\*\*\*Esse medida tem variações para os tipos Inteiro e Real.**

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPIRITO SANTO

**Escala de unidade da informação: Múltiplos padronizados do bit.**

**BYTE (Conjunto de 8 bits): Na Linguagem de Programação, a estrutura de Dados Vetor (Rodar o Programa: Matrícula) é:**

PAUTA		
ALUNO	MATRÍCULA	ENDEREÇO DE MEMÓRIA
ALUNO 1	INTEIRO	HEXADECIMAL
ALUNO 2	INTEIRO	HEXADECIMAL
ALUNO 3	INTEIRO	HEXADECIMAL
...	....	.....
ALUNO 48	INTEIRO	HEXADECIMAL
ALUNO 49	INTEIRO	HEXADECIMAL
ALUNO 50	INTEIRO	HEXADECIMAL

Exemplo de armazenamento em memória de Estruturas de dados: VETOR (Array).

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

**Escala de unidade da informação: Múltiplos padronizados do bit.**

**UNICODE:** A sigla *Universal Coded Character Set*, corresponde a uma codificação de caracteres de **largura variável**, de 1 até 4 bytes com até 8 bits (no máximo) por byte, gerando uma sequência de padrões denominados de Formato de Transformação Unicode (ou *Unicode Transformation Format* - UTF).

Number of bytes	Bits for code point	First code point	Last code point	Byte 1	Byte 2	Byte 3	Byte 4
1	7	U+0000	U+007F	0xxxxxxx			
2	11	U+0080	U+07FF	110xxxxx	10xxxxxx		
3	16	U+0800	U+FFFF	1110xxxx	10xxxxxx	10xxxxxx	
4	21	U+10000	U+10FFFF	11110xxx	10xxxxxx	10xxxxxx	10xxxxxx

Exemplo variável do Padrão UTF – 8 estratificado por bytes.

# Introdução à Computação usando Python

## A unidade da informação




### Múltiplos padronizados do bit: UTF – 8.

	_0	_1	_2	_3	_4	_5	_6	_7	_8	_9	_A	_B	_C	_D	_E	_F
0_	NUL 0000	SOH 0001	STX 0002	ETX 0003	EOT 0004	ENQ 0005	ACK 0006	BEL 0007	BS 0008	HT 0009	LF 000A	VT 000B	FF 000C	CR 000D	SO 000E	SI 000F
1_	DLE 0010	DC1 0011	DC2 0012	DC3 0013	DC4 0014	NAK 0015	SYN 0016	ETB 0017	CAN 0018	EM 0019	SUB 001A	ESC 001B	FS 001C	GS 001D	RS 001E	US 001F
2_	SP 0020	! 0021	" 0022	# 0023	\$ 0024	% 0025	& 0026	' 0027	( 0028	) 0029	* 002A	+ 002B	, 002C	- 002D	. 002E	/ 002F
3_	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	: 003A	; 003B	< 003C	= 003D	> 003E	? 003F
4_	@ 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
5_	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[ 005B	\ 005C	] 005D	^ 005E	_ 005F
6_	` 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
7_	p 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	 007C	} 007D	~ 007E	DEL 007F
8_	• +00	• +01	• +02	• +03	• +04	• +05	• +06	• +07	• +08	• +09	• +0A	• +0B	• +0C	• +0D	• +0E	• +0F
9_	• +10	• +11	• +12	• +13	• +14	• +15	• +16	• +17	• +18	• +19	• +1A	• +1B	• +1C	• +1D	• +1E	• +1F
A_	• +20	• +21	• +22	• +23	• +24	• +25	• +26	• +27	• +28	• +29	• +2A	• +2B	• +2C	• +2D	• +2E	• +2F
B_	• +30	• +31	• +32	• +33	• +34	• +35	• +36	• +37	• +38	• +39	• +3A	• +3B	• +3C	• +3D	• +3E	• +3F
2_	2 0000	2 0000	LATIN 0000	LATIN 0000	LATIN 0100	LATIN 0140	LATIN 0180	LATIN 01C0	LATIN 0200	IPA 0240	IPA 0280	IPA 02C0	ACCENTS 0300	ACCENTS 0340	GREEK 0380	GREEK 03C0
2_	CYRIL 0400	CYRIL 0440	CYRIL 0480	CYRIL 04C0	CYRIL 0500	ARMENI 0540	HEBREW 0580	HEBREW 05C0	ARABIC 0600	ARABIC 0640	ARABIC 0680	ARABIC 06C0	SYRIAC 0700	ARABIC 0740	THAANA 0780	N'KO 07C0
3_	INDIC 0800	MISC. 1000	SYMBOL 2000	KANA... 3000	CJK 4000	CJK 5000	CJK 6000	CJK 7000	CJK 8000	CJK 9000	ASIAN A000	HANGUL B000	HANGUL C000	HANGUL D000	PUA E000	FORMS F000
4_	SMP... 10000	□ 40000	□ 80000	SSP... C0000	SPU... 100000	4 140000	4 160000	4 180000	5 200000	5 2200000	5 2400000	5 2600000	6 2800000	6 3000000		

### UTF – 8.

### Características do UTF – 8:

- ✓ Tamanho variável.
- ✓ Predomina na WEB.
- ✓ Recomendada pela [W3C](http://www.w3.org) (  ) para XML e HTML.
- ✓ Atualização permanente com os padrões do universo WEB.
- ✓ Exemplo atual: [W3C Workshop on](http://www.w3.org/2003/01/22-xml-Unicode/)

[Web Games.](http://www.w3.org/2003/01/22-xml-Unicode/)



# Introdução à Computação usando Python

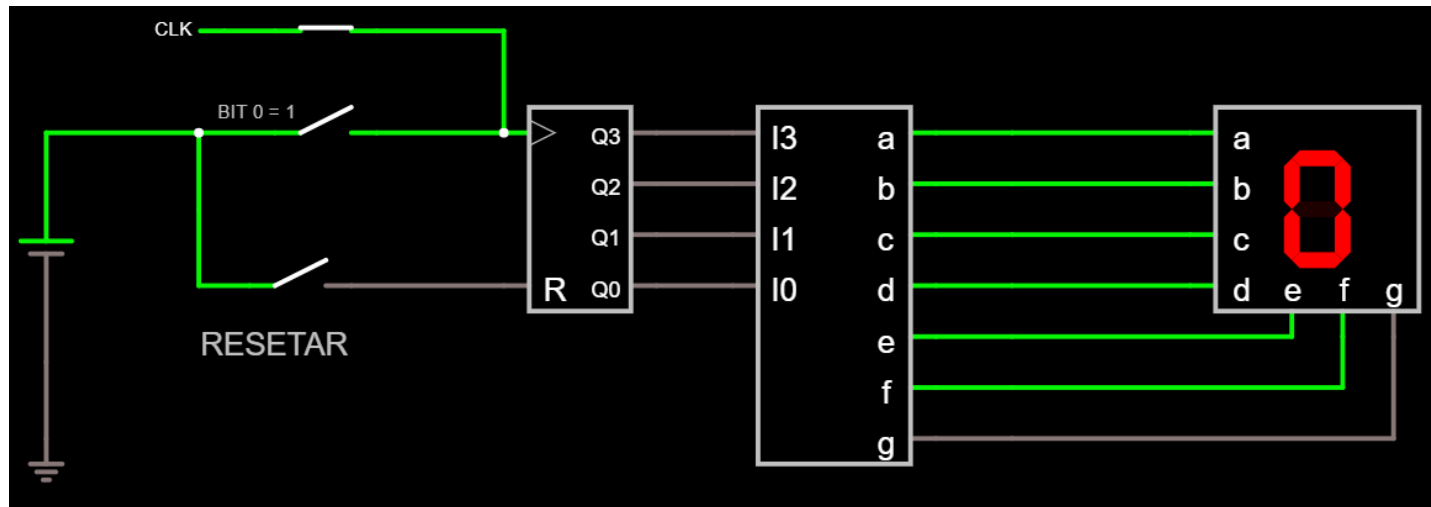
## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPIRITO SANTO

**Escala de unidade da informação: Múltiplos padronizados do bit.**

**Padrão WEB:** É a codificação do padrão UTF na base 16 (Base Hexadecimal) para memorização dos diversos códigos de comunicação.



BASE HEXADECIMAL CONTÉM 16 SÍMBOLOS: 0 ate 9 + A até F.

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPIRITO SANTO

Sistema Numérico Binário x Decimal x Hexadecimal:

<b>BIT 3 = 8</b>	<b>BIT 2 = 4</b>	<b>BIT 1 = 2</b>	<b>BIT 0 = 1</b>	<b>BASE 10</b>	<b><u>BASE 16</u></b>
0	0	0	0	0	0
0	0	0	1	1	1
0	0	1	0	2	2
0	0	1	1	3	3
0	1	0	0	4	4
0	1	0	1	5	5
0	1	1	0	6	6
0	1	1	1	7	7
1	0	0	0	8	8
1	0	0	1	9	9
1	0	1	0	10	A
1	0	1	1	11	B
1	1	0	0	12	C
1	1	0	1	13	D
1	1	1	0	14	E
1	1	1	1	15	F



# Introdução à Computação usando Python

## A unidade da informação

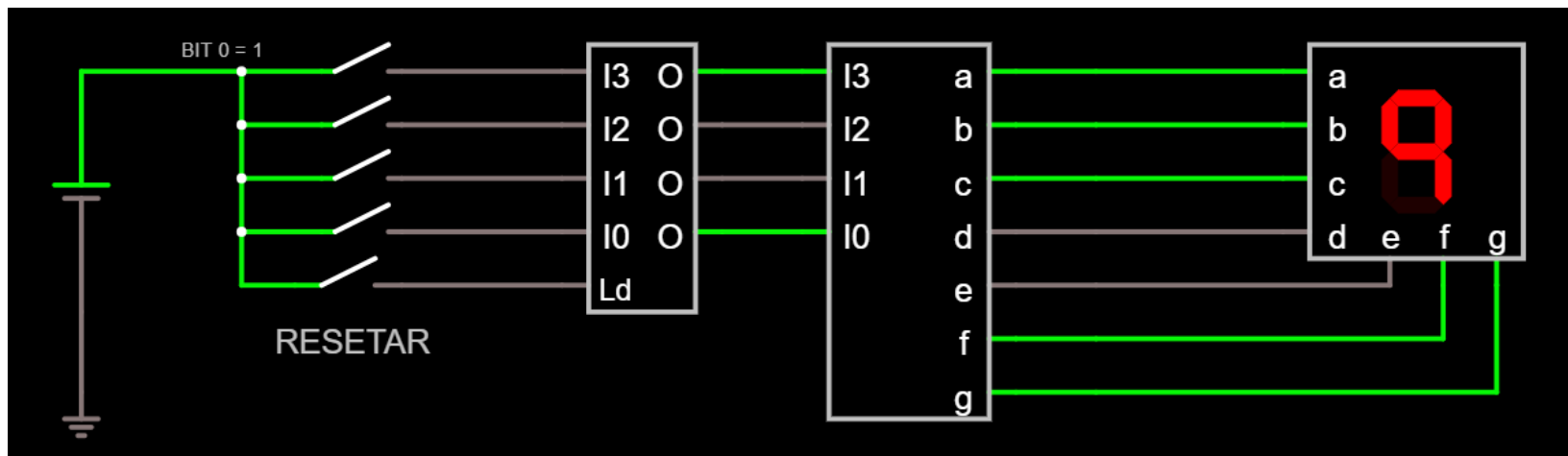


UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

**Escala de unidade da informação: Múltiplos padronizados do bit.**

**PADRÃO DE 8 BITS = 1 BYTE = CODIFICADO POR 16 SÍMBOLOS.**

**16 SÍMBOLOS = BASE HEXADECIMAL: ENDEREÇO DE MEMÓRIA.**



**REGISTRADOR: BASE HEXADECIMAL CONTÉM 16 SÍMBOLOS: 0 ate 9 + A até F.**

# Introdução à Computação usando Python

## A unidade da informação

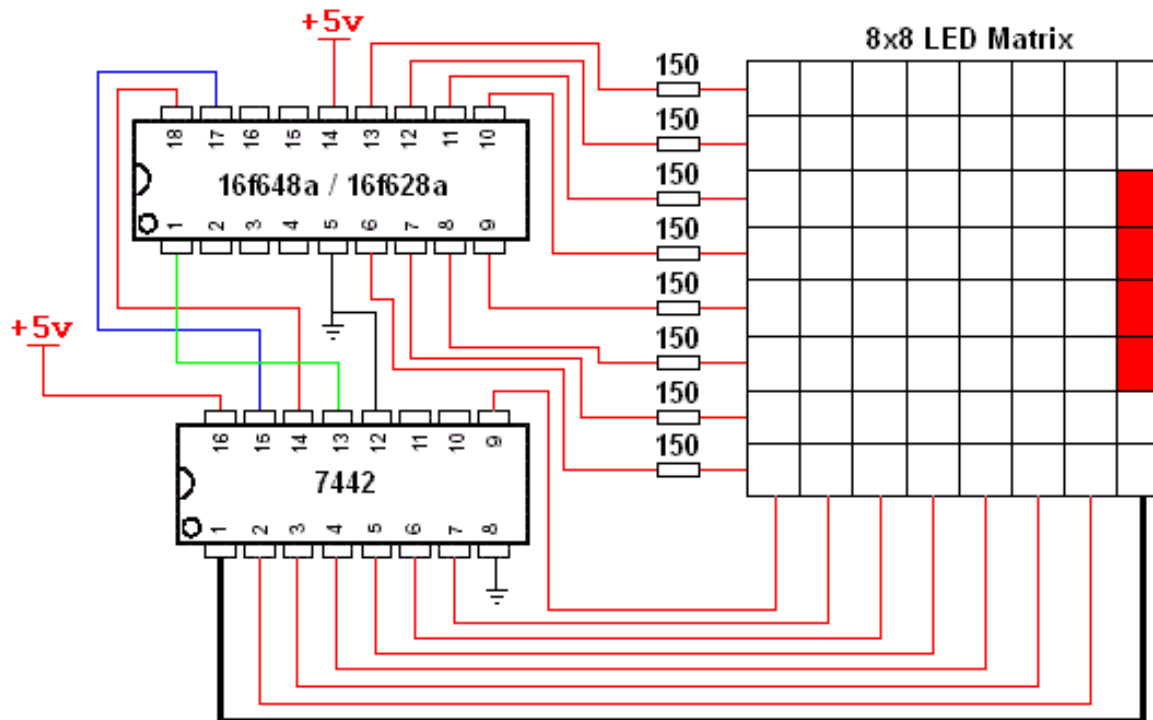


UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

**Escala de unidade da informação: Múltiplos padronizados do bit.**

**PADRÃO DE 8 BITS = 1 BYTE = CODIFICADO POR 16 SÍMBOLOS.**

**16 SÍMBOLOS = BASE HEXADECIMAL: ENDEREÇO DE MEMÓRIA.**



Matriz (Array) de 8 x 8 LEDs.

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

Como fazer a mudança de **Base 10 para Base 16** ?

Faça a sequência algorítmica a seguir:

**PASSO 1:** Dividir o número sucessivas vezes por 16.

**PASSO 2:** Parar a divisão em um resto  $\leq 15$ .

**PASSO 3:** Juntar os números da Direita para Esquerda fazendo a substituição dos números:

- 10 pela letra = A
- 11 pela letra = B
- 12 pela letra = C
- 13 pela letra = D
- 14 pela letra = E
- 15 pela letra = F

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

Como fazer a mudança de **Base 10 para Base 16** ?

Exemplos:

$$25_{10} = 19_{16}$$

25		16	
9		1	
		PARAR	
		JUNTAR AO CONTRÁRIO:	19

$$26_{10} = 1A_{16}$$

26		16	
10 = A		1	
		PARAR	
		JUNTAR AO CONTRÁRIO:	1A

$$43_{10} = 2B_{16}$$

43		16	
11 = B		2	
		PARAR	
		JUNTAR AO CONTRÁRIO:	2B

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

Como fazer a mudança de **Base 10 para Base 16** ?

Exemplos:

$$448_{10} = 1C0_{16}$$

448	16			
	28	16		
0	12 = C	1	JUNTAR AO CONTRÁRIO:	1C0
		PARAR		

$$2575_{10} = A0F_{16}$$

2575	16			
	160	16		
15 = F	0	10 = A	JUNTAR AO CONTRÁRIO:	A0F
		PARAR		

$$43_{10} = 2B_{16}$$

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPIRITO SANTO

Transformação Hexadecimal x Decimal: Exemplos de mudança de base.

BASE 16			BASE 16	BASE 10
$16^2$	$16^1$	$16^0$		
<b>0</b>	<b>1</b>	<b>9</b>	$0 \times 16^2 + 1 \times 16^1 + 9 \times 16^0 = 0 \times 256 + 1 \times 16 + 9 \times 1 = 0 + 16 + 9 = 25$	<b>25</b>
<b>0</b>	<b>A</b>	<b>2</b>	$0 \times 16^2 + A \times 16^1 + 2 \times 16^0 = 0 \times 256 + A \times 16 + 2 \times 1 = 0 + 160 + 2 = 162$	<b>162</b>
<b>0</b>	<b>A</b>	<b>F</b>	$0 \times 16^2 + A \times 16^1 + F \times 16^0 = 0 \times 256 + A \times 16 + F \times 1 = 0 + 160 + 15 = 175$	<b>175</b>
<b>A</b>	<b>0</b>	<b>F</b>	$A \times 16^2 + 0 \times 16^1 + F \times 16^0 = 10 \times 256 + 0 \times 16 + F \times 1 = 2560 + 0 + 15 = 2575$	<b>2575</b>
<b>1</b>	<b>C</b>	<b>0</b>	$1 \times 16^2 + C \times 16^1 + 0 \times 16^0 = 1 \times 256 + C \times 16 + 0 \times 1 = 256 + 192 + 0 = 448$	<b>448</b>

Como fazer a mudança de Base 16 para Base 10 ?

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPIRITO SANTO

Transformação Hexadecimal x Decimal: Exemplos de mudança de base.

BASE 16			BASE 16	BASE 10
$16^2$	$16^1$	$16^0$		
<b>0</b>	<b>1</b>	<b>9</b>	$0 \times 16^2 + 1 \times 16^1 + 9 \times 16^0 = 0 \times 256 + 1 \times 16 + 9 \times 1 = 0 + 16 + 9 = 25$	<b>25</b>
<b>0</b>	<b>A</b>	<b>2</b>	$0 \times 16^2 + A \times 16^1 + 2 \times 16^0 = 0 \times 256 + A \times 16 + 2 \times 1 = 0 + 160 + 2 = 162$	<b>162</b>
<b>0</b>	<b>A</b>	<b>F</b>	$0 \times 16^2 + A \times 16^1 + F \times 16^0 = 0 \times 256 + A \times 16 + F \times 1 = 0 + 160 + 15 = 175$	<b>175</b>
<b>A</b>	<b>0</b>	<b>F</b>	$A \times 16^2 + 0 \times 16^1 + F \times 16^0 = 10 \times 256 + 0 \times 16 + F \times 1 = 2560 + 0 + 15 = 2575$	<b>2575</b>
<b>1</b>	<b>C</b>	<b>0</b>	$1 \times 16^2 + C \times 16^1 + 0 \times 16^0 = 1 \times 256 + C \times 16 + 0 \times 1 = 256 + 192 + 0 = 448$	<b>448</b>

Como fazer a mudança de Base 16 para Base 10 ?

# Introdução à Computação usando Python

## A unidade da informação



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

## SISTEMA INTERNACIONAL DE UNIDADES

Prefixos do Sistema Internacional de Unidades - SI							
Prefixo		1000 <sup>m</sup>	10 <sup>n</sup>	Escala curta	Escala longa	Equivalente numérico	Desde <sup>[nota 1]</sup>
Nome	Simbolo						
<b>iota</b>	Y	1000 <sup>8</sup>	10 <sup>24</sup>	Septilhão	Quadrilhão	1 000 000 000 000 000 000 000 000	1991
<b>zeta</b>	Z	1000 <sup>7</sup>	10 <sup>21</sup>	Sextilhão	Milhar de trilhão	1 000 000 000 000 000 000 000	1991
<b>exa</b>	E	1000 <sup>6</sup>	10 <sup>18</sup>	Quintilhão	Trilhão	1 000 000 000 000 000 000	1975
<b>peta</b>	P	1000 <sup>5</sup>	10 <sup>15</sup>	Quadrilhão	Milhar de bilhão	1 000 000 000 000 000	1975
<b>tera</b>	T	1000 <sup>4</sup>	10 <sup>12</sup>	Trilhão	Bilhão	1 000 000 000 000	1960
<b>giga</b>	G	1000 <sup>3</sup>	10 <sup>9</sup>	Bilhão	Milhar de milhão	1 000 000 000	1960
<b>mega</b>	M	1000 <sup>2</sup>	10 <sup>6</sup>	Milhão	Milhão	1 000 000	1960
<b>quilo</b>	k	1000 <sup>1</sup>	10 <sup>3</sup>	Mil	Milhar	1 000	1795
<b>hecto</b>	h	1000 <sup>2/3</sup>	10 <sup>2</sup>	Cem	Centena	100	1795
<b>deca</b>	da	1000 <sup>1/3</sup>	10 <sup>1</sup>	Dez	Dezena	10	1795
<i>nenhum</i>		1000 <sup>0</sup>	10 <sup>0</sup>	Unidade	Unidade	1	
<b>deci</b>	d	1000 <sup>-1/3</sup>	10 <sup>-1</sup>	Décimo	Décimo	0,1	1795
<b>centi</b>	c	1000 <sup>-2/3</sup>	10 <sup>-2</sup>	Centésimo	Centésimo	0,01	1795
<b>mili</b>	m	1000 <sup>-1</sup>	10 <sup>-3</sup>	Milésimo	Milésimo	0,001	1795
<b>micro</b>	μ	1000 <sup>-2</sup>	10 <sup>-6</sup>	Milionésimo	Milionésimo	0,000 001	1960
<b>nano</b>	n	1000 <sup>-3</sup>	10 <sup>-9</sup>	Bilionésimo	Milésimo de milionésimo	0,000 000 001	1960
<b>pico</b>	p	1000 <sup>-4</sup>	10 <sup>-12</sup>	Trilionésimo	Bilionésimo	0,000 000 000 001	1960
<b>femto</b>	f	1000 <sup>-5</sup>	10 <sup>-15</sup>	Quadrilionésimo	Milésimo de bilionésimo	0,000 000 000 000 001	1964
<b>atto</b>	a	1000 <sup>-6</sup>	10 <sup>-18</sup>	Quintilionésimo	Trilionésimo	0,000 000 000 000 000 001	1964
<b>zepto</b>	z	1000 <sup>-7</sup>	10 <sup>-21</sup>	Sextilionésimo	Milésimo de trilionésimo	0,000 000 000 000 000 000 001	1991
<b>iocto</b>	y	1000 <sup>-8</sup>	10 <sup>-24</sup>	Septilionésimo	Quadrilionésimo	0,000 000 000 000 000 000 000 001	1991



# Introdução à Computação usando Python

## A unidade da informação

---



UNIVERSIDADE  
**VILA VELHA**  
ESPÍRITO SANTO

### Questionário de Teste: Google Forms.



FAVOR INSERIR NO FORMATO:

SEU NOME <SEU E-MAIL>

EXEMPLO:

FULANO DE TAL <FULANODETAL@GMAIL.COM>

Aprendendo a usar o **Google Forms**.