

Codificações

Paulo Ricardo Lisboa de Almeida



Codificações

Quando representamos letras, palavras, números ... por um **grupo especial de símbolos**, estamos criando uma **codificação**.

Cada letra tem seu próprio grupo, que podemos chamar de código.

Exemplo de codificação?

Codificações

Quando representamos letras, palavras, números ... por um **grupo especial de símbolos**, estamos criando uma **codificação**.

Cada letra tem seu próprio grupo, que podemos chamar de código.

Exemplo de codificação: Código Morse.

Codificações

Precisamos de codificações para representar caracteres, letras especiais, símbolos, ... em binário.

ASCII

Uma das formas mais simples de representações é o código **ASCII**.

American **S**tandard **C**ode for **I**nformation **I**nterchange.

Código de 7 bits.

Quantos códigos diferentes temos?

ASCII

Uma das formas mais simples de representações é o código **ASCII**.

American **S**tandard **C**ode for **I**nformation **I**nterchange.

Código de 7 bits.

$$2^7 = 128 \text{ códigos.}$$

Representar todos os caracteres do seu teclado, além de códigos de controle (e.g., quebra de linha '\n').

Dica

Digite `man ascii` no terminal para ver a tabela ASCII.

Exemplo

Um exemplo em C.

```
int main(){  
    char c1 = 'a'; //caractere a  
    char c2 = '9'; //caractere 9  
  
    //...  
  
    return 0;  
}
```


Exemplo

Um exemplo em C.

```
int main(){  
    char c1 = 'a'; //caractere a  
    char c2 = '9'; //caractere 9  
  
    //...  
  
    return 0;  
}
```

← 0110 0001 armazenado na memória

Exemplo

Um exemplo em C.

```
int main(){  
    char c1 = 'a'; //caractere a  
    char c2 = '9'; //caractere 9 ← 0011 1001 armazenado na memória  
  
    //...  
  
    return 0;  
}
```

Note que o caractere 9 (0011 1001) tem um valor diferente do número 9_{10} (1001_2)

Observação

Nossas máquinas são **comumente** endereçadas a byte.

Cada endereço de memória suporta **exatamente** 1 byte.

Você aprenderá detalhes na disciplina de Arquitetura de Computadores.

Sendo assim, comumente um caractere ASCII vai ocupar 1 byte, e não 7 bits.

Os valores padrão sempre vão começar com um 0 em binário.

O bit extra é muitas vezes utilizado para se criar extensões da tabela ASCII.

Contendo por exemplo caracteres específicos de determinados alfabetos, como o 'ç'.

Ou para utilização de um sistema de paridade.

Detecção de erros.

ASCII

O código ASCII foi um dos primeiros padrões a serem adotados em larga escala.

Possibilitou que as máquinas se comuniquem.

Se cada máquina utiliza sua própria codificação, fica difícil a comunicação entre elas,

ASCII

A tabela ASCII foi criada para que conversões sejam feitas de maneira simples e rápida.

Converter de ASCII para 0-9

Decimal	ASCII	Binário
0	011 0000	0
1	011 0001	1
2	011 0010	10
3	011 0011	11
4	011 0100	100
5	011 0101	101
6	011 0110	110
7	011 0111	111
8	011 1000	1000
9	011 1001	1001

Como converter de binário para ASCII, e vice-versa?

Converter de ASCII para 0-9

Decimal	ASCII	Binário
0	011 0000	0
1	011 0001	1
2	011 0010	10
3	011 0011	11
4	011 0100	100
5	011 0101	101
6	011 0110	110
7	011 0111	111
8	011 1000	1000
9	011 1001	1001

Basta ligar/desligar os bits indicados!

Podemos usar uma operação lógica, o que pode ser mais rápido (veremos adiante) ou somar/subtrair 48_{10}

Maiúsculo/Minúsculo

a : 01100001	A : 01000001
b : 01100010	B : 01000010
c : 01100011	C : 01000011
d : 01100100	D : 01000100
e : 01100101	E : 01000101

...

...

x : 01111000	X : 01011000
y : 01111001	Y : 01011001
z : 01111010	Z : 01011010

Como converter entre maiúsculo ou
minúsculo?

Maiúsculo/Minúsculo

a : 01100001	A : 01000001
b : 01100010	B : 01000010
c : 01100011	C : 01000011
d : 01100100	D : 01000100
e : 01100101	E : 01000101
...	...
x : 01111000	X : 01011000
y : 01111001	Y : 01011001
z : 01111010	Z : 01011010

Basta ligar/desligar os bits indicados!
Podemos usar uma operação lógica, o que
pode ser mais rápido ou somar/subtrair 32.

Note que $32 = 2^5$.

Outras Codificações

Existem diversas outras codificações que utilizamos nos nossos dia a dia.

Exemplos:

Compatíveis com o ASCII (adicionam novos códigos, mas são compatíveis com o ASCII).

UTF8;

CP1252.

Código BCD.

Para nos comunicarmos com dispositivos elétricos simples.

Gray Code.

Observações

Se você está comunicando dois dispositivos, uma das coisas que você deve levar em consideração é a codificação.

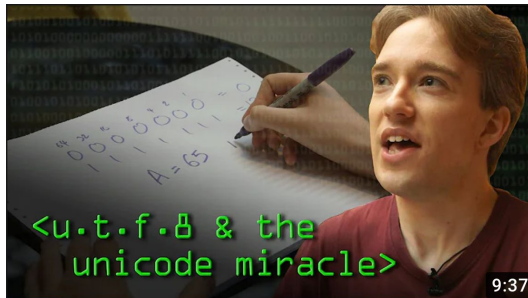
Exemplo: um dispositivo utiliza BCD para representar números, e outro ASCII.

Precisa de conversão.

Exercícios

1. Assista a esse vídeo:

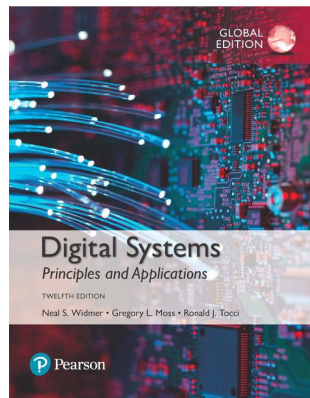
<https://youtu.be/MijmeoH9LT4>



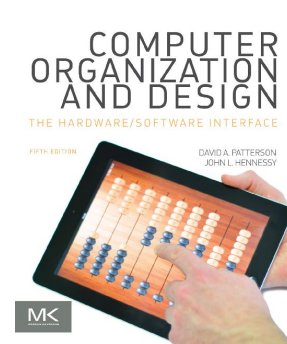
2. Pesquise sobre UTF-8. Como ele é formado? Quantos bits ocupa? Como ele é compatível com ASCII? Quem usa essa codificação? E o CP1252, quem usa? Que problemas são causados pelo uso dessas diferentes codificações?

Referências

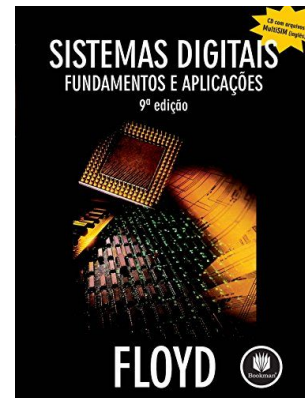
Ronald J. Tocci, Gregory L. Moss, Neal S. Widmer. Sistemas digitais. 10a ed. 2017.



Hennessy, J. L., Patterson, D. A. Computer Organization and Design: The Hardware/Software Interface. 2014.



Thomas Floyd. Widmer. Sistemas Digitais: Fundamentos e Aplicações. 2009.



Licença

Esta obra está licenciada com uma Licença [Creative Commons Atribuição 4.0 Internacional](https://creativecommons.org/licenses/by/4.0/).