Sistema binário

Existem duas maneiras de representar uma informação: analogicamente ou digitalmente. Uma música é gravada numa fita K-7 de forma analógica, codificada na forma de uma grande onda de sinais magnéticos, que pode assumir um número ilimitado de frequências. Um som grave seria representado por um ponto mais baixo da onda, enquanto um ponto mais alto representaria um som agudo. O problema com esta representação, é que qualquer interferência causa distorções no som. Se os computadores trabalhassem com dados analógicos, certamente seriam muito passíveis de erros, pois qualquer interferência, por mínima que fosse, causaria alterações nos dados processados e consequentemente nos resultados.

O sistema digital por sua vez, permite armazenar qualquer informação na forma de uma sequencia de valores positivos e negativos, ou seja, na forma de uns e zeros. O número 181, por exemplo, pode ser representado digitalmente como 10110101. Qualquer tipo de dado, seja um texto, uma imagem, um vídeo, um programa, ou qualquer outra coisa, será processado e armazenado na forma de uma grande sequencia de uns e zeros.

Cada valor binário é chamado de "bit", contração de "binary digit" ou "dígito binário". Um conjunto de 8 bits forma um byte, e um conjunto de 1024 bytes forma um Kilobyte (ou Kbyte). O número 1024 foi escolhido, pois é a potência de 2 mais próxima de 1000. Um conjunto de 1024 Kbytes forma um Megabyte (1048576 bytes) e um conjunto de 1024 Megabytes forma um Gigabyte (1073741824 bytes). Os próximos múltiplos são o Terabyte (1024 Gibabytes) e o Petabyte (1024 Terabytes).

Também usamos os termos Kbit, Megabit e Gigabit, para representar conjuntos de 1024 bits. Como um byte corresponde a 8 bits, um Megabyte corresponde a 8 Megabits e assim por diante.

1 Bit =	1 ou 0
1 Byte =	Um conjunto de 8 bits
1 Kbyte =	1024 bytes ou 8192 bits
1 Megabyte =	1024 Kbytes, 1.048.576 bytes ou 8.388.608 bits
1 Gigabyte =	1024 Megabytes, 1.048.576 Kbytes, 1.073.741.824 bytes ou 8.589.934.592 bits

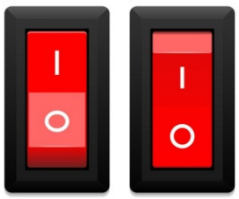
Quando vamos abreviar, também existe diferença. Quando estamos falando de Kbytes ou Megabytes, abreviamos respectivamente como KB e MB, sempre com o "B" maiúsculo. Quando estamos falando de Kbits ou Megabits abreviamos da mesma forma, porém usando o "B" minúsculo, "Kb", "Mb" e assim por diante. Parece irrelevante, mas esta é uma fonte de muitas confusões. Sempre que nos referimos à velocidade de uma rede de computadores, por exemplo, não a medimos em bytes por segundo, e sim em bits por segundo: 10 megabits, 100 megabits e assim por diante. Escrever "100 MB" neste caso daria a entender que a rede transmite a 100 megabytes, que correspondem a 800 megabits.

É justamente o uso do sistema binário que torna os computadores confiáveis, pois a possibilidade de um valor 1 ser alterado para um valor 0, o oposto, é muito pequena. Lidando com apenas dois valores diferentes, a velocidade de processamento também se torna maior, devido à simplicidade dos cálculos.

Usando a operação de apenas dois dígitos ou estados da álgebra booleana (sim ou não, verdadeiro ou falso, ligado ou desligado e 0 ou 1, por exemplo), o sistema binário permite que os computadores processem dados com maior efetividade.

Qualquer valor diferente desses dois algarismos será desprezado pela máquina, fato que promove maior confiabilidade aos cálculos. Isso é uma grande vantagem em relação aos mecanismos que processam informações de maneira analógica, os quais são mais suscetíveis a “ruídos” (em outras palavras, distorções na transmissão de dados).

Aliado à lógica booleana, o sistema binário permite representar números, caracteres ou símbolos, e realizar operações lógicas ou aritméticas por meio de circuitos eletrônicos digitais (também chamados de portas lógicas).



O sistema operacional do computador identifica as combinações numéricas através do valor positivo ou negativo aplicado pelo programador aos zeros e uns do programa em execução. Assim, a leitura dos códigos binários funciona como um interruptor: quando o computador identifica o 1, a luz acende; ao se deparar com o 0, a luminosidade é apagada (são feitas milhares de leituras por segundo!).

Por meio desses sinais, a máquina pode realizar os cálculos e processamentos necessários para transformar o conteúdo codificado em um formato que possamos compreender – seja texto, imagem ou som.

Todos os softwares são codificados e armazenados com base no sistema binário. Isso significa que, se pudéssemos abrir o disco rígido do computador e ler o que está escrito nele, veríamos uma lista, aparentemente, interminável de zeros e uns.

Representação de caracteres

ASCII E CP-437

Os primeiros computadores foram projetados somente para usuários da língua inglesa. O alfabeto inglês, possui 26 letras. Considerando que o computador diferencia letras maiúsculas de minúsculas, então, são necessários 52 símbolos para representar todas as letras, 10 símbolos para os números indo-arábicos e outro tanto para caracteres como símbolos gráficos, símbolos de pontuação, símbolos matemáticos e símbolos de controle (não imprimíveis).

Todavia, o computador somente entende números binários (0 e 1). Deste modo, é necessário relacionar um número binário com a representação gráfica de um caractere formando uma tabela de símbolos. Esse tipo de tabela recebe o nome de página de código (code page).

A primeira iniciativa para padronizar uma página de código foi feita pela indústria telegráfica americana e foi chamada de codificação ASCII - American Standard Code for Information Interchange - Código Americano Para Intercâmbio de Informações. A primeira versão da tabela ASCII utilizava codificação base 2 com sete dígitos (2 ^ 7 = 128 caracteres binários). Por exemplo, o código ASCII binário para a letra "A" (maiúscula) com sete dígitos é representado como 1000001.

A tabela ASCII foi estendida com a codificação base 2, agora em oito dígitos. O código para a letra "A" (maiúscula) de um byte é 01000001. No sistema decimal, o zero à esquerda não representa nada. Entretanto, no sistema binário, ao acrescentar um digito à esquerda, aumentamos exponencialmente a capacidade de representação do número binário. Um binário com oito dígitos pode representar 256 caracteres, pois 2 ^ 8 = 256. Os processadores Intel 8008, Motorola 6800 e Zilog Z80, são todos chips com arquitetura de 8 bits utilizados na construção dos primeiros computadores pessoais.

A página de código CP-437 da IBM (chamada de MS-DOS Latin) foi uma das primeiras tentativas de internacionalização, como afirma. Essa página de código internacional utilizava o alfabeto latino inglês, adicionado a uma série de caracteres acentuados utilizados nos principais países da Europa ocidental.

Podemos destacar os seguintes caracteres acentuados da página CP-437:

Do espanhol (Á, Í, Ó, Ú).

Do francês (À, Â, È, Ê, Ë, Ì, Î, Ï, Ô, Œ, œ, Ù, Û).

Do português (Á, À, Â, Ã, ã, Ê, Í, Ó, Ô, Õ, õ, Ú).

A internacionalização se deu dentro de um contexto de forte globalização e de neoliberalismo presente nos anos 1980-90. Não havia consenso na indústria de computação e cada fabricante adotava seu próprio padrão de página de código. Isso levou a uma verdadeira torre de babel com o uso de padrões proprietários e padrões abertos que não se comunicavam ou se comunicavam com muita dificuldade.

ISO-8859-N, ANSI E WINDOWS-1252

A maior tentativa de normatizar o processo de internacionalização foi idealizada pelo consórcio ISO/IEC e recebeu o nome de padrão ISO-8859-N. Esse padrão gerou uma série de páginas de código em oito bits. À medida que a indústria da computação avançava sobre a Europa oriental (devido à queda do muro de Berlim, em 1989), surgia a necessidade de atualizar as páginas de código.

É possível obter uma relação de países e suas respectivas páginas de código. O padrão ISO-8859-N possui uma página de código para cada alfabeto com caracteres comuns em vários países.

Por exemplo, ISO-8859-1 (chamado também de Latin 1) é a página de código padrão para países que falam as seguintes línguas: dinamarquês, holandês, finlandês, francês, alemão, islandês, irlandês, italiano, norueguês, português, espanhol, catalão e sueco (falado na Finlândia por Linus Torvalds).

No total, foram definidas 16 páginas de código (ISO-8859-1 até 8859-16) que possuem em comum o fato de serem todas codificadas em 8 bits e terem os caracteres do alfabeto latino. Outra curiosidade sobre o ISO-8859-1, é que esse é o conjunto definido para ser utilizado por uma invenção recente da época: World Wide Web. ISO-8859 também é usado como um dos padrões para codificação de tipos MIME ainda hoje.

A Microsoft adotou um "padrão proprietário" criando um super conjunto semelhante ao ISO-8859-n chamado de Windows-1252, chamado também de código ANSI. Os sistemas Windows-1252 e ISO-8859, apesar de semelhantes, têm muitas diferenças técnicas e não se comunicam facilmente.

Em meados da década de 1990, depois que a Europa estava dominada pela indústria americana da computação, os esforços de internacionalização se voltaram para o oriente. Os mercados do Japão, da Coréia e da China passaram a ser o maior objetivo comercial dessas empresas em função de seu tamanho e poder de compra.

A questão sobre a internacionalização em países asiáticos, é que eles não utilizam caracteres latinos, como na maior parte do ocidente. Os países asiáticos, de modo geral, utilizam um conjunto de símbolos gráficos para representar palavras, nomes, conceitos e ideias. Por exemplo, o Kanji (hanzi) é a escrita formal japonesa e possui aproximadamente 4.000 símbolos. Nesse tipo de escrita, um símbolo pode representar um ou mais conceitos ou pode ter mais de um símbolo para o mesmo conceito. Assim, na escrita do oriente não encontramos o conceito de letras ou sílabas como conhecemos no ocidente.

Além disso, existe a questão da direção da leitura. Nas línguas latinas, em geral, a leitura é feita da esquerda para a direita e de cima para baixo. Em línguas orientais encontramos também a leitura da direita para a esquerda, e em alguns casos, de baixo para cima. Podendo haver até mesmo uma variação em ambas as direções ao mesmo tempo dentro do mesmo texto (raro).

No oriente são utilizados ideogramas (conceito semelhante ao da escrita egípcia) e os logogramas (símbolos gráficos) que juntos somam milhares de símbolos gráficos. Essa quantidade enorme de caracteres não pode ser representada em uma página de código de 8 bits.

A solução foi representar os caracteres orientais em um sistema com 16 bits ou dois bytes - 2 ^ 16 = 65.536 caracteres. Dois bytes formando um único caractere é um tipo computacional chamado Word.

As codificações ISO-8859 e Windows-1252 tiveram seu mérito. Entretanto, o péssimo suporte para caracteres orientais e o fato de que sistemas proprietários são incompatíveis com sistemas abertos, acabaram levando a uma nova tentativa de internacionalização.

UNICODE E UTF-8

O consórcio Unicode é uma entidade sem fins lucrativos que coordena o desenvolvimento de uma página de código universal. Essa super página de códigos terá os símbolos de todos os alfabetos existentes, atuais ou extintos, desde que tenham relevância para a comunidade científica. Unicode pretende substituir todos os sistemas de páginas de código existentes no mundo e se tornar o único sistema utilizado para esse fim.

Na prática, isso significa que a página de código Unicode teria mais de 110 mil símbolos no total. Isso inclui alfabetos atuais, de escritas extintas, de línguas indígenas, de antigos alfabetos de sinais como hieróglifos egípcios, runas inglesas e até a escrita cuneiforme dos

sumérios. Além disso, são incluídos todos os sinais matemáticos e os símbolos musicais utilizados para escrever partituras e muitos outros símbolos e gráficos estranhos ou raros.

Obviamente, a iniciativa é um exagero desmedido, já que a maioria das pessoas utiliza apenas o alfabeto de sua própria língua. Poucas pessoas escrevem textos multilíngues e, quando o fazem, usam de duas até quatro línguas normalmente similares. Manter milhares de caracteres em memória para uso eventual, não é viável em termos de computação. O padrão Unicode é uma daquelas boas ideias que na teoria são ótimas, mas na prática não dão certo.

Assim, surgiu a ideia de implementar um sistema compatível com Unicode, mas que usasse apenas os caracteres necessários. Esse sistema foi chamado de UTF - Unicode Transformation Format. UTF é um método utilizado para implementar Unicode usando menos recursos de memória e disco. Existem várias versões de UTF como UTF-8, UTF-16 e UTF-32. No momento, a que tem maior suporte e permite melhor uso é UTF-8.

UTF-8 é flexível e utiliza de um até quatro bytes para representar um caractere. Em Unicode, os caracteres são sempre representados como U+0000, onde um 0 representa um byte. Se utilizarmos apenas um byte podemos representar até 256 caracteres como no sistema ASCII. Por exemplo, de U+0000 até U+007F podemos representar os 128 caracteres do ASCII legado e de (U+0080 até U+00FF podemos representar os 128 caracteres do ASCII estendido).

Na prática, os caracteres ocidentais e orientais podem ser representados com dois bytes apenas já que 2 ^ 16 = 65.536 símbolos. Os problemas são alguns símbolos que utilizam três ou até quatro bytes, tornando UTF-8 tão complexo quanto o próprio Unicode. A migração dos sistemas ANSI e ISO-8859-N para UTF-8 não está completa e alguns programas podem apresentar sérios problemas durante a migração (principalmente bancos de dados). Apesar disso, parece que UTF-8 será o padrão de codificação de caracteres em vários sistemas operacionais dentro de pouco tempo.

TECLADO JAPONÊS

Resolvido o problema da exibição de milhares de caracteres com o uso de páginas de código de dois bytes, agora era preciso um método que permitisse usar um teclado padrão para inserir caracteres gráficos em editores de texto adaptados para línguas orientais. O teclado padrão possui um número limitado de possibilidades. Mesmo que uma tecla possa representar até 4 símbolos, o número de combinações possíveis é inferior ao total de símbolos necessários para usar a escrita do dia-a-dia em países orientais.

Por exemplo, conhecendo 2.000 símbolos em Kanji (hanzi), é possível ler mais de 90% de um jornal, um livro ou uma revista escritos nesse formato. Esse é o número aproximado de símbolos que são ensinados durante o ensino médio no Japão. Ler no Japão é uma questão de classe social. Quanto mais símbolos você conhece, maior sua posição social e mais formal sua escrita. A quantidade total de símbolos Kanji pode chegar a mais de 4.000 (carece de fonte confiável).

Os países que utilizam símbolos como palavras, usam o teclado do computador de forma bem distinta dos países ocidentais. Enquanto digitamos letras para formar sílabas (que formam palavras), os japoneses digitam a representação gráfica dos fonemas ou silabogramas. Esses sinais são a representação gráfica do som da palavra tal como ela é soletrada. O fonema é o som de uma sílaba quando uma palavra é pronunciada. Existem

dois silabários utilizados para representar fonemas do idioma Kanji: o hiragana e o katakana. Eles são chamados comumente de kanas.

Podemos afirmar, grosso modo, que o Kanji é o japonês formal (escrito com milhares de símbolos) enquanto os kanas representam o japonês falado (silabogramas) com aproximadamente cinquenta símbolos cada um.

Os kanas podem ser utilizados isoladamente, como um sistema de escrita, ou como um complemento ao Kanji. Isso ocorre quando não há um símbolo Kanji para representar o que se quer dizer, neste caso se usa um kana.

Ao todo, são quatro os sistemas de escrita vigentes na língua japonesa:

- Kanji: de origem chinesa e coreana, tem mais de 2.000 anos, representa ideias, verbos, nomes de pessoas, coisas e lugares. Um único símbolo pode significar várias coisas diferentes. É a língua culta do Japão, falada e escrita, conta atualmente com mais de 4.000 símbolos em uso e vários em desuso.
- Hiragana: caracteres fonéticos (representam sons) ou silabários são utilizados quando não há um Kanji próprio para definir um conceito. Os símbolos hiragana são utilizados como um método de escrita atualmente no Japão.
- Katakana: caracteres fonéticos (representam sons) ou silabários usados para escrever palavras originadas de outros idiomas, principalmente do francês e do inglês que são as línguas que mais influenciaram o japonês moderno.
- Romaji: caracteres latinos (como os do alfabeto português) utilizados para representar siglas estrangeiras como ONU, NASA ou OTAN. Também é utilizado para representar nomes de cidadãos japoneses em passaportes, cartões de visita ou documentos internacionais. Os romaji são utilizados por marcas registradas, nomes de empresas ou de produtos japoneses vendidos fora do Japão. São exemplos de escrita romaji marcas como SONY, Panasonic, AIWA, Honda, Yamaha e Toyota. A escrita romaji é uma forma de "romanização" de fonemas japoneses, sendo considerada também uma forma de transliteração.

HIRAGANA

O hiragana é composto por 48 símbolos que representam os fonemas do japonês moderno. Esse tipo de linguagem, onde cada símbolo representa um fonema, é chamado de silabograma. O conjunto de silabogramas hiragana é formado por:

- 5 vogais (a, i, u, e, o) - na ordem japonesa!

- 9 consoantes (w, r, y, m, h, n, t, s, k) que obrigatoriamente são combinadas com as 5 vogais formando 42 silabogramas (as combinações yi, ye, e wu não são utilizadas). As combinações wi, e we são obsoletas no hiragana moderno.
- O fonema wo atualmente é usado apenas como uma partícula.
- O "n" é uma consoante no singular e não se associa com nenhuma vogal.

O quadro a seguir, lista os 48 fonemas hiragana.

Utilizando o alfabeto hiragana, é possível representar os 48 fonemas no teclado padrão de computador. O teclado japonês é do tipo QWERTY com um layout quase idêntico ao teclado US Internacional, considerando apenas as letras latinas utilizadas na escrita romaji.

Cada um dos 48 símbolos hiragana está representados no teclado, juntamente com as letras latinas do romaji, números indo-arábicos, pontuação, sinais gráficos e matemáticos. É possível alternar entre hiragana e katakana apertando uma tecla. Mas os símbolos katakana não estão estampados no teclado sendo preciso conhecê-los.

Observe o layout do teclado japonês:

Por exemplo, o número um (1) em japonês tem o som ['/ichi'/] ('chi' tem som de '/tchi'/) então, basta consultar na tabela hiragana quais são os fonemas que devem ser digitados (i + chi). O editor de textos deve ser configurado para fazer a transliteração. Assim, se houver um ou mais símbolos Kanji para o fonema o editor oferecerá as opções para que o usuário escolha uma dentre aquelas encontradas em seu banco de dados IME - "Input Method Editor".

Se não houver um símbolo Kanji, será mantida a escrita hiragana ou katakana. Existem casos em que os kanas são mantidos mesmo que haja um Kanji correspondente. Esse é um recurso para maior compreensão do leitor e ocorre quando um Kanji raro, ou em desuso, for apresentado como opção.

Referências utilizadas:

- https://www.tecmundo.com.br/infografico/9424-como-um-computador-faz-calculos-pelo-sistema-binario-.htm
- http://www.hardware.com.br/livros/hardware-manual/como-funciona-sistema-binario.html
- https://www.vivaolinux.com.br/artigo/Internacionalizacao-de-Caracteres-em-Computadores

---

Atividades

1. Fazer um resumo deste texto.
2. Explique como funciona o sistema binário no computador.
3. Como o computador interpreta os caracteres?
4. Na tabela ASCII, quais os códigos decimais para os seguintes caracteres:
    a. B
    b. ?
    c. 5
    d. m