

Código de barras

Arquivo fonte: `barcode.{c | cpp | java}`

Autor: Antonio Cesar de Barros Munari (Fatec Sorocaba)

Os códigos de barra são um grande avanço tecnológico e organizacional para a vida moderna, pois permitem uma agilidade muito grande na identificação dos produtos a serem manipulados nas várias operações do dia a dia de uma empresa. Existem algumas variações na forma de codificação, mas a ideia é basicamente a mesma: um prefixo numérico chamado *System Number*, o *Manufacturer Code*, o *Product Code* e o *Check Digit*. No Brasil e em muitos países do mundo é utilizada a versão EAN-13 dos códigos de barra, que será apresentada a seguir.



Fig. 1: Exemplo de um código de barras no formato EAN-13

Na figura 1 temos um exemplo de código de barras no formato explorado neste problema. O código total é composto por 13 dígitos numéricos, o que explica em parte o nome EAN-13. Os dois primeiros dígitos são os chamados números do sistema (no caso, 76); os próximos 5 dígitos formam o chamado *Manufacturer Code* (que no exemplo é 12345); os 5 dígitos seguintes compõem o chamado *Product Code* (67890 em nosso exemplo) e o dígito restante é o chamado *Check Digit* (o dígito verificador, que é calculado com base nos 12 primeiros dígitos e que em nosso exemplo é 0). Toda etiqueta de código de barras possui o código representado em duas formas: a forma numérica decimal, que aparece na parte inferior da figura, e a forma gráfica em colunas, que é aquele conjunto de linhas verticais característico desse tipo de código visando a leitura rápida por meio de dispositivos óticos especiais. O leitor ótico reconhece as colunas escuras da etiqueta como sendo o valor 1 e as colunas brancas como sendo o valor 0. Repare na figura que o primeiro dígito não é expresso por colunas, ao contrário dos dígitos restantes. Temos inicialmente um conjunto de três colunas que correspondem a um delimitador padrão obrigatório chamado *Left Guard*: uma coluna escura, uma coluna clara, uma coluna escura, que é entendida pelo leitor como sendo 101. Em seguida temos seis dígitos expressos por meio de arranjos de 7 colunas cada, que correspondem ao segundo dígito do número do sistema e aos 5 dígitos do *Manufacturer Code*. Depois existe um novo delimitador padrão obrigatório, chamado *Central Guard*, que é formado por uma coluna clara, uma escura, uma clara, uma escura e outra clara, que é reconhecida pelo leitor como sendo a sequência 01010. Então aparecem os 6 dígitos finais, cada um expresso por um arranjo de 7 colunas, que correspondem aos 5 dígitos do *Product Code* e ao dígito verificador. A codificação se encerra com o *Right Guard*, que é composto por uma coluna escura, uma clara e outra escura, formando a sequência 101.

Então, cada dígito numérico decimal do código corresponde a uma combinação de zeros e uns de comprimento 7. Para determinar qual a representação de zeros e uns de cada dígito, precisamos levar em conta uma série de fatores. O primeiro deles é em que lado do *Central Guard* está o dígito: se estiver antes desse delimitador, ou seja, do seu lado esquerdo, a representação tem um critério, se o dígito estiver depois do delimitador, ou seja, do seu lado direito, a representação tem outro critério. Para os dígitos do lado esquerdo, precisamos saber inicialmente o valor do primeiro dígito, aquele que é o primeiro número do sistema e que não aparece nas colunas. A Tabela 1 indica como o segundo, terceiro, quarto, quinto e sexto dígitos devem ser codificados em barras, de acordo com o valor desse primeiro dígito. Já para os dígitos do lado direito, a codificação é sempre a mesma, como mostra a Tabela 2.

| Primeiro dígito | Díg. 2 | Díg. 3 | Díg. 4 | Díg. 5 | Díg. 6 | Díg. 7 |
|-----------------|--------|--------|--------|--------|--------|--------|
| 0 | Ímpar | Ímpar | Ímpar | Ímpar | Ímpar | Ímpar |
| 1 | Ímpar | Ímpar | Par | Ímpar | Par | Par |
| 2 | Ímpar | Ímpar | Par | Par | Ímpar | Par |
| 3 | Ímpar | Ímpar | Par | Par | Par | Ímpar |
| 4 | Ímpar | Par | Ímpar | Ímpar | Par | Par |
| 5 | Ímpar | Par | Par | Ímpar | Ímpar | Par |
| 6 | Ímpar | Par | Par | Par | Ímpar | Ímpar |
| 7 | Ímpar | Par | Ímpar | Par | Ímpar | Par |
| 8 | Ímpar | Par | Ímpar | Par | Par | Ímpar |
| 9 | Ímpar | Par | Par | Ímpar | Par | Ímpar |

Tabela 1: Mapa de paridade dos dígitos do lado esquerdo do código de barras EAN-13

| Dígito | Lado Esquerdo | | Lado Direito |
|--------|----------------|--------------|--------------|
| | Paridade Ímpar | Paridade Par | |
| 0 | 0001101 | 0100111 | 1110010 |
| 1 | 0011001 | 0110011 | 1100110 |
| 2 | 0010011 | 0011011 | 1101100 |
| 3 | 0111101 | 0100001 | 1000010 |
| 4 | 0100011 | 0011101 | 1011100 |
| 5 | 0110001 | 0111001 | 1001110 |
| 6 | 0101111 | 0000101 | 1010000 |
| 7 | 0111011 | 0010001 | 1000100 |
| 8 | 0110111 | 0001001 | 1001000 |
| 9 | 0001011 | 0010111 | 1110100 |

Tabela 2: Representação em colunas dos dígitos decimais do código de barras

O código de exemplo 7612345678900 teria então a seguinte codificação em colunas: 101 (*Left Guard*) 0101111 (o ‘6’ que corresponde ao primeiro conjunto de colunas) 0110011 (o ‘1’ que corresponde ao segundo conjunto de colunas) 0010011 (‘2’) 0100001 (‘3’) 0100011 (‘4’) 0111001 (‘5’) 01010 (*Central Guard*) 1010000 (‘6’) 1000100 (‘7’) 1001000 (‘8’) 1110100 (‘9’) 1110010 (‘0’) 1110010 (‘0’) 101 (*Right Guard*). Juntando tudo temos:

```
1010101111011001100100110100001010001101110010101010100001000100100100
0111010011100101110010101
```

Isso quer dizer que se um leitor ótico lesse a etiqueta da Figura 1, geraria exatamente essa sequência de zeros e uns.

O dígito verificador, por sua vez, é calculado de uma maneira muito simples: somam-se os dígitos decimais localizados nas posições ímpares do código. Em nosso exemplo, seria $7 + 1 + 3 + 5 + 7 + 9 = 32$. Depois geramos um segundo total, pela soma dos dígitos das posições pares (menos o último dígito, que é o que queremos calcular). Em nosso exemplo seria $6 + 2 + 4 + 6 + 8 + 0 = 26$. Adicionamos à primeira soma o triplo da segunda, o que em nosso exemplo corresponde a $32 + 3 * 26 = 32 + 78 = 110$. O dígito verificador é o módulo da diferença entre esse número e o primeiro múltiplo de 10 maior ou igual a esse número. Em nosso exemplo produzimos o valor 110, que por coincidência, é também múltiplo de 10. Então o dígito verificador procurado é $110 - 110 = 0$, ou seja, o *Check Digit* para o código do exemplo da Figura 1 é 0.

Sua tarefa neste problema é, dado um conjunto de leituras de código de barras realizada por um leitor ótico, verificar se o dígito verificador está correto ou não.

Entrada

A entrada é composta por vários casos de teste, cada um expresso por uma linha contendo uma sequência de 95 caracteres zeros e uns que corresponde à leitura do reconhecedor ótico. A entrada é sinalizada pelo final de arquivo.

Saída

Para cada caso de teste, imprima uma linha com a mensagem ‘barcode incorreto: lido = N esperado = X ’, caso o dígito verificador esteja incorreto. Nesse caso N representa o dígito verificador lido na entrada e X o dígito verificador correto para aquele código de barras. A mensagem deverá estar totalmente em minúsculas. Caso o dígito verificador esteja correto, imprimir o número do sistema, o *Manufacturer Code*, o *Product Code* e o dígito verificador, separados por um hífen, como mostram os exemplos.

Exemplos

Entrada:

```
1010101111011001100100110100001010001101110010101010100001000100100100
0111010011100101110010101
1010111011000100100010010110001010000100010110101011100101110010100111
0100001010011101101100101
1010001011010011101100110010011011110100111010101011001101101100100001
0101110010011101000100101
101001001100011010100111010011100011010101011100101110010100001
0100010010001001001110101
```

Saída:

```
76-12345-67890-0
97-88539-00535-2
59-01234-12345-7
barcode incorreto: lido = 5 esperado = 4
```