

Codificação Gamma de Elias

Contest Local, Universidade de Ulm  Alemanha**Timelimit: 1**

O código gamma de Elias é um código simples que pode ser usado para codificar uma sequência de inteiros positivos. Iremos utilizar um código modificado que também é capaz de codificar zeros. Para codificar um inteiro n , faça o seguinte:

1. Seja k o numero de bits de n
2. Escreva $k-1$ zeros seguido por 1
3. Escreva n em binário

Exemplos

Número	Binário	Número de bits	Prefixo	Código
0	0	1	1	10
1	1	1	1	11
2	10	2	01	0110
3	11	2	01	0111
4	100	3	001	001100
5	101	3	001	001101
6	110	3	001	001110
7	111	3	001	001111
8	1000	4	0001	00011000

Uma sequência de inteiros é codificada escrevendo os códigos dos inteiros individuais da sequência na mesma ordem em que os inteiros aparecem na sequência. O prefixo de k bits a mais, antes da representação binária de cada inteiro, é necessário para poder decodificar os inteiros codificados. Então, quando estiver lendo a codificação de uma sequência de inteiros, se lermos $k-1$ zeros seguido por um, isso significa que existem k bits seguintes, que são a representação binária do próximo inteiro codificado.

Se quisermos diminuir o tamanho da codificação da sequência de inteiros, pode haver ainda algum espaço para melhorias vamos considerar as duas otimizações seguintes:

1. Se houver um prefixo que indica os k bits seguintes, mas se não tiver um inteiro na sequência com k bits, podemos usar este prefixo para indicar que seguem $k+1$ bits. Se já houver um prefixo que indica que seguem $k+1$ bits, esse prefixo não será mais necessário e poderá ser usado para indicar que seguem $k+2$ bits, e assim por diante.
2. Podemos adiciona um zero a esquerda da representação binária de todos os inteiros na sequência com k bits, que então torna-se inteiros com $k+1$ bits, e então a primeira otimização pode ser usada. A otimização parece especialmente útil se houver alguns inteiros com k bits, mas muitos inteiros com mais de k bits.

Quando estivermos diminuindo o tamanho da codificação da sequência de inteiros, nós apenas devemos ter

cuidado sobre quantos inteiros na sequência tem um certo número de bits. Seja c_i o número de inteiro na sequência com i bits.

Vejamos o seguinte exemplo $c_1 = 2$, $c_2 = 4$, $c_3 = 0$, $c_4 = 1$ (que, por exemplo, poderia corresponder a sequência 2, 1, 3, 8, 0, 2, 3). Com o original código gamma de elias, a codificação da sequência deveria ter tamanho $2 \times (1 + 1) + 4 \times (2 + 2) + 0 \times (3 + 3) + 1 \times (4 + 4) = 28$. Usando a otimização 1 podemos salvar 1 bit usando o prefixo 001 para o inteiro com 4 bits. Então, poderíamos usar a otimização 2 e adicionar zeros a esquerda do inteiro com 1 bit, fazendo-os usar 2 bits. Então, usamos a otimização 1 e o prefixo 1 para inteiros com 2 bits, prefixo 01 para inteiros com 4 bits, e teremos o novo tamanho de $6 \times (1 + 2) + 1 \times (2 + 4) = 24$.

Ambas otimizações podem ser usadas várias vezes. O objetivo é combinar essas duas otimizações da melhor maneira possível, o que significa que queremos encontrar uma codificação de uma determinada sequência de inteiros que tem um tamanho mínimo entre todas as codificações usando a Codificação Gamma de Elias com qualquer combinação dessas duas otimizações.

Entrada

A entrada contém vários casos teste. Cada casos teste inicia com uma linha contendo um inteiro n , ($1 \leq n \leq 128$). A próxima linha contém os c_1 , ..., c_n ($0 \leq c_i \leq 10000$). A entrada termina com $n = 0$.

Saída

Para cada caso teste imprima uma linha como o tamanho mínimo de uma codificação da determinada sequência da entrada.

Exemplo de Entrada	Exemplo de Saída
4 2 4 0 1 5 9 4 2 4 3 11 44 56 96 26 73 80 77 50 33 16 78 0	24 99 5494