Problem name: Tickets

Language: English

Solution

This problem is ad-hoc problem where in order to come up with the solution we need to carefully analyze the given method for creating the gray code.

The solution for 30 points is to simulate described method and print the K^{th} string in the list. Both time and space complexity of this solution is $O(2^n)$.

The solution for 100 points is a little bit more complicated. First, we can see that if $K \leq 2^{n-1}$ (if the number that we are looking for is in the first half of the gray code), then the first bit in the result is 0, and if $K > 2^{n-1}$, then the solution starts with 1. Furthermore, now when we have found the first bit in the solution, let's erase the first bit from all numbers in the gray code. What we are left with is an array where the first half is the gray code of all numbers consisting of $n-1$ bits, and the second half is same as the first half, just reversed. So if we are looking for $K \leq 2^{n-1}$ then what we are left with is the problem of finding the K^{th} element in the gray code of all numbers consisting of $n-1$ bits, and if $K > 2^{n-1}$ then we need to find the $(K - 2^{n-1})^{th}$ element in the reversed gray code of order $n - 1$, which is the same as searching for $(2^n - K + 1)^{th}$ element in the gray code of order $n - 1$. To get all the bits, we repeat this process n times.

So, the algorithm for 100 points is:

```
While $(n > 0)$
  If $K \leq 2^{n-1}$ print 0
  Else /* $K > 2^{n-1}$ */
    print 1
    $K = 2^n - K + 1$
  $n = n - 1$
```

The time complexity of this solution is $O(n)$ and space complexity is constant.