

# C1 I题题解

---

题目懒得复制粘贴了，直接题解。

首先我们需要知道一个结论：

$$4k \oplus 4k + 1 \oplus 4k + 2 \oplus 4k + 3 = 0$$

我们注意到这四个数其二进制表示仅有最后两位不同，分别为00,01,10,11，其按位异或值恰为0；而其余位皆相同，按位异或值为0，故结论成立。

我们将依据此结论构造数组。注意到，每四个这样的数其异或值为0，在多个数异或的情况下，我们可以尝试四个四个分组直到只剩几个落单的数。我们也可以从4这个数切入此题。

我们先将 $n$ 分为4种情况，分别为 $4i$ ， $4i + 1$ ， $4i + 2$ ， $4i + 3$ ，每种情况从理论最小值 $n-1$ 往上考虑。每次考虑一个值时，我们可以先算出从0到最大数 $max$ 的异或值 $p$ （在四个四个分组的情况下，我们只需要算不多于4个数的异或值），再删去 $max + 1 - n$ 个数，并使得这些删去的数异或值为 $p$ 。这将问题转化为：从小于最大数 $max$ 的数中选出 $max + 1 - n$ 个数，并使这些数异或值为 $p$ ，一定程度上简化了问题。

下面来具体分情况讨论：

- $n = 4i$

考虑最小值 $4i - 1$ ，此时四个四个分组，得到的异或值恰为0，故答案为 $n-1$ 。

- $n = 4i + 3$

考虑最小值 $4i + 2$ ，此时四个四个分组，仅剩下 $4i$ ， $4i + 1$ ， $4i + 2$ ，其异或值为 $4i + 3$ ，(可以通过上结论反推)，故不成立。

再考虑 $4i + 3$ ，此时0到 $4i + 3$ 的异或值恰为0，我们需要删除一个数，并使其异或值为0，删除0即可，故答案为 $n$ 。

- $n = 4i + 1$

考虑最小值 $4i$ ，此时四个四个分组，得到的异或值为 $4i$ ，故不成立。

再考虑 $4i + 1$ ，此时0到 $4i + 1$ 分组后仅剩下 $4i$ ， $4i + 1$ ，其异或值为1，我们需要删除一个数，并使其异或值为1，删除1即可，故答案为 $n$ 。

- $n = 4i + 2$

考虑最小值 $4i + 1$ ，此时四个四个分组，仅剩下 $4i$ ， $4i + 1$ ，其异或值为1，故不成立。

再考虑 $4i + 2$ ，此时0到 $4i + 2$ 的异或值为 $4i + 3$ ，我们需要删除一个数，并使其异或值为 $4i + 3$ ，这是不可能实现的（当然不可能删除还没出现的 $4i + 3$ ），故不成立。

再考虑 $4i + 3$ ，此时0到 $4i + 3$ 的异或值为0，我们需要删除两个数，并使其异或值为0，这是不可能实现的（因为只有两个相同的数异或值为0），故不成立。

再考虑 $4i + 4$ ，此时0到 $4i + 4$ 的异或值为 $4i + 4$ ，我们需要删除三个数，并使其异或值为 $4i + 4$ ，我们考虑 $4i + 4$ 的二进制表示，首位为1，其后有若干1和0，我们可以考虑以下两种情况：

- a. 二进制表示首位之后有至少一个1，我们可以将其拆分为三个数：0，首位为1其后均为0，其余位数。具体地，比如1000101，可拆分为0,1000000,101，此时保证这三个数均小于 $4i + 4$ ，互不相同，且异或值为 $4i + 4$ 。此时答案为 $n+2$ 。
- b. 二进制表示首位1后全是0，我们无法处理其首位的1，因为只有 $4i + 4$ 在这一位为1，但我们不能删除 $4i + 4$ ，故不成立。

对这种子情况，再考虑 $4i + 5$ ，此时0到 $4i + 5$ 的异或值为1，我们需要删除四个数，并使其异或值为1，我们无法在一组里取四个数（这样异或为0），考虑到前面数位的问题，我们可以尝试在两组里分别取两个数，仅考虑二进制表示最后两位，直接枚举，我们容易发现取出 $4j, 4j + 2, 4k, 4k + 3, j \neq k$ ，异或为1，满足条件。此时答案为 $n+3$ 。

实现代码如下：

```
#include<bits/stdc++.h>
using namespace std;
int solve(int x)
{
    int res=0;
    while(x>0)
    {
        x>>=1;
        res++;
    }
    return res;
}
int main()
{
    int t,n;
    // freopen("1.in","r",stdin);
    // freopen("1.out","w",stdout);
    scanf("%d",&t);
    while(t--)
```

```
{
    scanf("%d",&n);
    if(n%4==0)
    {
        printf("%d\n",n-1);
    }
    else if(n%4==3)
    {
        printf("%d\n",n);
    }
    else if(n%4==1)
    {
        printf("%d\n",n);
    }
    else if(n%4==2)
    {
        if(solve(n+2)>solve(n))
        {
            printf("%d\n",n+3);
        }
        else
        {
            printf("%d\n",n+2);
        }
    }
}
}
```

*Author:*张家源