Permutation

胡晋侨

得分情况

- 大概过了不到一半？（没仔细看）

30%

- 直接返回随机值，多交几次

30%-60%

- 本来是指望有选手想出正确率略大于1/2，又没那么接近1的算法
- 但实际考试中变成了欧皇测试

100%

- 做法不唯一
- std只进行了三次询问，正确率$\geq 1-2^{-31}$

- 考虑$b=1$时P有哪些性质可以利用
- 难点在于x_2的值会被随机函数mask

- 有$x_1 \oplus x_3 = f_2(x_2)$
- 以及$x_2 = x_0 \oplus f_1(x_1) = x_4 \oplus f_3(x_3)$

100%

- 首先任取x_0, x_1，得到$x_3 \circ x_4 = P(x_0 \circ x_1)$
- 然后任取$v \neq 0$，令$y_0 = x_0 \oplus v; y_1 = x_1; z_4 = x_4 \oplus v; z_3 = x_3$
- 询问得到$y_3 \circ y_4 = P(y_0 \circ y_1), z_0 \circ z_1 = P^{-1}(z_3 \circ z_4)$
- 不难注意到，此时有$y_2 = z_2 = v \oplus x_2$
- 于是必然有$y_1 \oplus y_3 = z_1 \oplus z_3$

100%

- 若$b = 0$，则以$1 - 2^{-64}$的概率$y_3 \circ y_4 \neq z_3 \circ z_4$
- 在这一条件下，由于z_1几乎是均匀随机，故$y_1 \oplus y_3 = z_1 \oplus z_3$发生的概率$\approx 2^{-32}$
- 总的出错概率$\leq 1 - 2^{-31}$

- 所以出错的概率可以忽略不计，只需要询问三次即可
- 如果不放心，也可以将上述过程重复若干次

~~100%~~

- 有一名选手（非集训队）貌似注意到$b = 0$和$b = 1$的情况下，$getperm$的执行时间不一样。。然后使用$clock()$通过了本题
- ~~（我大意了没有混淆运行时间）~~
- ~~（要怪就怪评测鸭太准了）~~

Trivia

- 本题是密码学中的经典结论，其中$b = 1$时构造的排列被称为深度为3的 $Feistel\ Network$
- 可以证明，如果只允许单向询问P，则任何（多项式时间）程序都无法以较好（显著>1/2）的概率区分$b = 0$与$b = 1$的情况
- 可以证明，若深度为4（即令 $x_5 = x_3 \oplus f_4(x_4)$，并输出 $x_4 \circ x_5$），或者更大，则即使允许双向询问P，也不可能以较好的概率区分$b = 0$与$b = 1$

- （这里的表述不太严谨）

Questions