为什么身份证最后一位可能出现 X（shí）？

2026-07-10　原文链接

网站版本 1c934d989f84 · 文章 ID h_2043b293a490ecc7

为什么身份证最后一位可能出现 X（shí）？

有些身份证号最后一位是 **X**。这里的 X 不读“叉”，而读“十”，表示数值 10。

身份证最后一位是校验码。它由前 17 位计算得到，用来发现常见录入错误，例如某一位输错、相邻两位输反等。

身份证校验的核心可以概括为一句话：

> 先选用模 11 作为校验环境，再用 2 的幂作为位置权重。

也就是把身份证号写成一个“广义二进制”加权和：

$N = \sum_{k=0}^{17} 2^k a_k$

再要求：

$N \equiv 1 \pmod{11}$

等价地：

$N - 1 \equiv 0 \pmod{11}$

这套设计解释了两个问题：
为什么要用模 11，为什么又偏偏用 2^k 作为权重。

---

1. 先确定模数：为什么是 11？

校验码的目的不是“算出一个好看的数字”，而是尽量发现录入错误。

最常见的错误有两类：

单个数字输错

例如：

$3 \to 8$

以及：

相邻两位输反

例如：

$37 \to 73$

要检测这些错误，模数的选择很关键。

2. 小于 10 的模数不够用

若模数 $m < 10$，数字之间可能相差 m。

例如使用 mod 9 时：

$0 \to 9$

数字差为：

9

于是：

$9 \equiv 0 \pmod 9$

无论这一位的权重是什么，都有：

$w \cdot 9 \equiv 0 \pmod 9$

也就是说，这类单个数字错误无法被发现。

所以，若希望稳定检测单个数字错误，模数至少不能小于 10。

---

3. mod 10 看似方便，但有硬伤

mod 10 的余数正好是：

$0, 1, 2, \cdots, 9$

如果用 mod 10，校验码可以全部写成数字，不会出现 X。

但 mod 10 不是质数：

$10 = 2 \cdot 5$

在模 10 下，存在非零数相乘后等于 0 的情况：

$2 \cdot 5 \equiv 0 \pmod{10}$

这会影响检错能力。

若某一位权重为 2，数字错差为 5，例如：

$1 \to 6$

加权和变化量为：

$2 \cdot 5 = 10 \equiv 0 \pmod{10}$

错误发生了，校验结果却不变。

---

4. 重新设计 mod 10 权重也难以兼顾换位错误

也许可以不用权重 2，重新设计一组 mod 10 权重。

但若要检测所有单个数字错误，每个权重 w_i 都必须与 10 互素。
否则存在非零数字差 d，使得：

$w_i d \equiv 0 \pmod{10}$

与 10 互素的余数只有：

$1, 3, 7, 9$

它们全是奇数。

再看相邻两位交换。设相邻权重为 w_i, w_{i+1}，相邻数字为 x, y。交换造成的变化量是：

$(w_i - w_{i+1})(y - x)$

如果 w_i, w_{i+1} 都是奇数，那么：

$w_i - w_{i+1}$

一定是偶数。

取一组相邻数字，使得：

$y - x = 5$

则：

$(w_i - w_{i+1})(y - x)$

必为 10 的倍数，于是：

$(w_i - w_{i+1})(y - x) \equiv 0 \pmod{10}$

这类相邻交换无法检测。

因此，mod 10 很难同时保证：

所有单个数字错误可检测

和：

所有相邻不同数字交换可检测

---

5. mod 11 是最小的合适选择

11 有两个优点。

第一，11 是质数。
在模 11 下，两个非零数相乘仍然非零。

第二，11 只比 10 多一个余数：

10

十进制字符只能表示：

$0, 1, \cdots, 9$

多出来的这个数值 10，只需要用一个额外符号表示。身份证中采用：

$X = 10$

若使用更大的模数，例如 mod 13，则需要额外表示：

$10, 11, 12$

至少要引入三个非数字字符。

所以，mod 11 是一个折中：
比 mod 10 更可靠，又只多付出一个 X 的字符成本。

---

6. 模数确定后，为什么不用十进制权重？

身份证号本身是十进制数字，最自然的想法是直接把它看成十进制数：

$M = \sum_{k=0}^{17} 10^k a_k$

然后做 mod 11 校验。

但十进制权重在 mod 11 下会退化。

因为：

$10 \equiv -1 \pmod{11}$

所以：

$10^k \equiv (-1)^k \pmod{11}$

于是：

$M = \sum_{k=0}^{17} 10^k a_k$

在 mod 11 下只剩：

$M \equiv a_0 - a_1 + a_2 - a_3 + \cdots + (-1)^{17} a_{17} \pmod{11}$

也就是说，权重只有两种：

$1, -1, 1, -1, \cdots$

这种校验本质上只区分奇数位和偶数位，位置信息保留得太少。

比如相隔两位的位置权重相同：

$10^k \equiv 10^{k+2} \pmod{11}$

如果第 k 位和第 $k+2$ 位交换，变化量是：

$(10^k - 10^{k+2})(y - x)$

而：

$10^k - 10^{k+2} \equiv 0 \pmod{11}$

所以这类交换不会改变校验结果。

因此，在已经选定 mod 11 的前提下，直接使用十进制权重并不理想。

---

7. 为什么选择 2^k：2 是 mod 11 的原根

身份证校验使用：

$N = \sum_{k=0}^{17} 2^k a_k$

关键原因是：

2

是模 11 的原根。

“原根” 可以简单理解为：
用它不断乘下去，在模 11 下可以生成所有非零余数。

具体看：

$2^0, 2^1, 2^2, \cdots, 2^9 \pmod{11}$

依次为：

$1, 2, 4, 8, 5, 10, 9, 7, 3, 6$

这正好是模 11 下的全部 10 个非零余数。

直到：

$2^{10} \equiv 1 \pmod{11}$

才开始循环。

所以，2 在模 11 下的周期是 10。
这说明 2^k 作为权重时，可以让不同位置尽量拥有丰富的非零权重。

对比十进制权重：

$10^k \pmod{11} = 1, -1, 1, -1, \cdots$

周期只有 2。

而二进制权重：

$2^k \pmod{11}$

周期为 10。

这就是为什么在 mod 11 校验中，不直接采用十进制权重，而采用 2^k 权重。

---

8. “转成二进制”是什么意思

这里说的“二进制”，不是把身份证号变成一串 0 和 1。

普通二进制数形如：

$\sum_{k=0}^{n} 2^k b_k$

其中每个 b_k 只能取 0 或 1。

身份证校验中使用的是：

$N = \sum_{k=0}^{17} 2^k a_k$

每个 a_k 仍然是十进制数字，校验位还允许 $X = 10$。

所以更准确地说，它是一个“广义二进制”加权和：
使用二进制的权重 2^k，但每一位的数字不局限于 $0, 1$。

---

9. 身份证校验公式

将身份证号从左到右写成：

$a_{17}a_{16}\cdots a_1 a_0$

其中 a_0 是最后一位校验码。

若最后一位是 X，规定：

$X = 10$

构造：

$N = a_0 + 2a_1 + 2^2 a_2 + \cdots + 2^{17} a_{17}$

即：

$N = \sum_{k=0}^{17} 2^k a_k$

身份证号码有效时满足：

$N \equiv 1 \pmod{11}$

也可以写成：

$N - 1 \equiv 0 \pmod{11}$

---

10. 常见权重表从哪里来

实际书写身份证号时，通常从左到右写 18 位。

最右边的校验位 a_0 权重是：

$2^0 = 1$

前 17 位从左到右对应权重为：

$2^{17}, 2^{16}, \cdots, 2^{1}$

对 11 取余后得到：

$7, 9, 10, 5, 8, 4, 2, 1, 6, 3, 7, 9, 10, 5, 8, 4, 2$

这就是身份证校验中常见的权重序列。

若前 17 位从左到右记为：

$b_1, b_2, \cdots, b_{17}$

对应权重为：

$w_1, w_2, \cdots, w_{17}$

则：

$S = \sum_{i=1}^{17} b_i w_i$

校验码记为 c，完整校验条件是：

$S + c \equiv 1 \pmod{11}$

因此：

$c \equiv 1 - S \pmod{11}$

---

11. X 从哪里来

模 11 的余数共有 11 种：

$0, 1, 2, \cdots, 10$

但单个十进制字符只有 10 个：

$0, 1, 2, \cdots, 9$

当校验码需要取数值 10 时，就要用一个额外符号表示。

身份证中规定：

$X = 10$

若：

$S \bmod 11 = r$

则：

$c \equiv 1 - r \pmod{11}$

对应关系如下：

$S \bmod 11$	校验码
0	1
1	0
2	X
3	9
4	8
5	7
6	6
7	5
8	4
9	3
10	2

所以，身份证最后一位出现 X，只是因为校验码需要表示数值 10。

---

12. 为什么能检测单个数字错误

假设某一位由 a 错输为 b，且：

$a \neq b$

这一位的权重为 w。
加权和变化量是：

$w(b-a)$

因为 a, b 都是十进制数字，所以：

$b-a \in \{\pm 1, \pm 2, \cdots, \pm 9\}$

在 mod 11 下，这些数都不是 0。

同时：

$w = 2^k$

而：

$2^k \not\equiv 0 \pmod{11}$

又因为 11 是质数，模 11 下两个非零数相乘仍非零：

$w(b-a) \not\equiv 0 \pmod{11}$

因此，任意一位数字输错，校验结果都会改变。

---

13. 为什么能检测相邻两位交换

设相邻两位原来是：

x, y

对应权重为：

$2^{k+1}, 2^k$

交换前贡献为：

$2^{k+1}x + 2^k y$

交换后贡献为：

$2^{k+1}y + 2^k x$

变化量为：

$2^{k+1}y + 2^k x - (2^{k+1}x + 2^k y)$

整理得：

$(2^{k+1} - 2^k)(y - x)$

即：

$2^k(y - x)$

若 $x \neq y$，则：

$y - x \in \{\pm 1, \pm 2, \cdots, \pm 9\}$

在 mod 11 下非零。

同时：

$2^k \not\equiv 0 \pmod{11}$

所以：

$2^k(y - x) \not\equiv 0 \pmod{11}$

因此，相邻两个不同数字交换，校验结果一定改变。

---

14. 小结

身份证校验的设计顺序可以这样理解：

1. 为了检测单个数字错误，模数不能小于 10；
2. mod 10 虽然不用 X，但不是质数，难以兼顾单错和相邻交换；
3. mod 11 是最小的可用质数，只比十进制多一个余数；
4. 余数 10 用 X 表示；

5. 确定 mod 11 后，十进制权重 10^k 会退化为 $1, -1$ 交替；

6. 2 是 mod 11 的原根，2^k 能生成全部非零余数；

7. 因此身份证采用广义二进制加权和：

$\sum_{k=0}^{17} 2^k a_k$

并用：

$\sum_{k=0}^{17} 2^k a_k \equiv 1 \pmod{11}$

进行校验。

身份证最后一位的 X，不是多余的复杂设计，而是用一个额外字符，换取更可靠的错误检测能力。