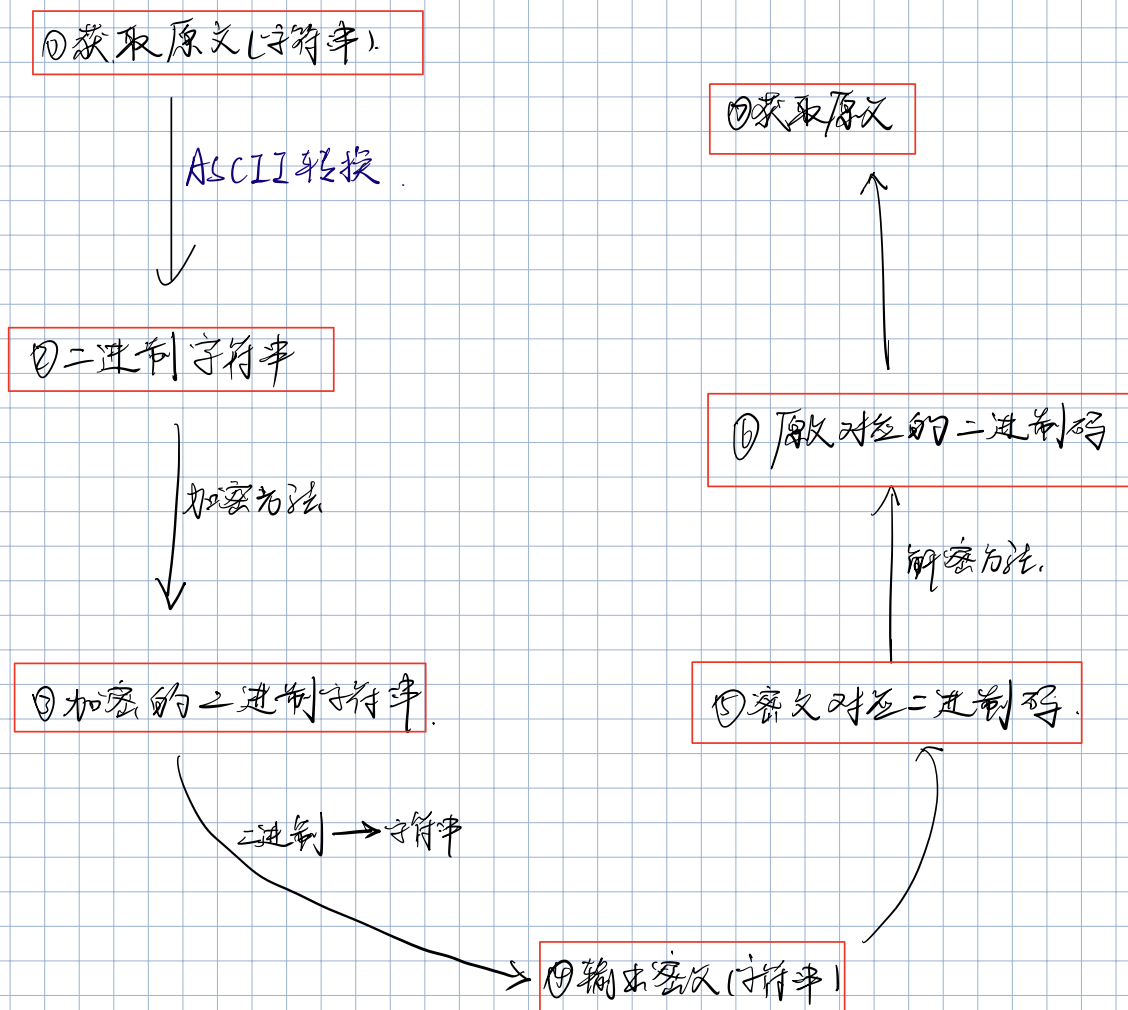
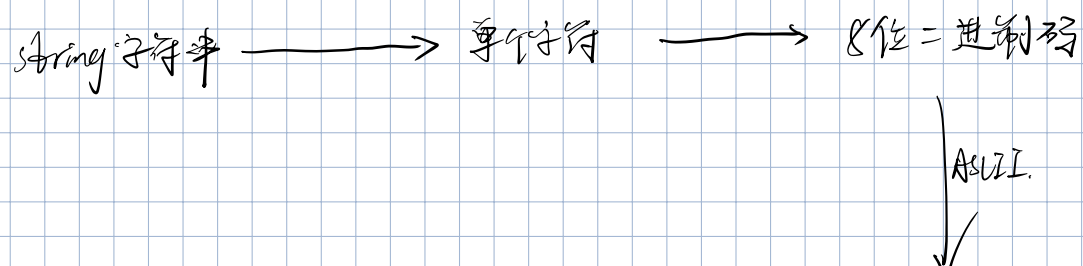


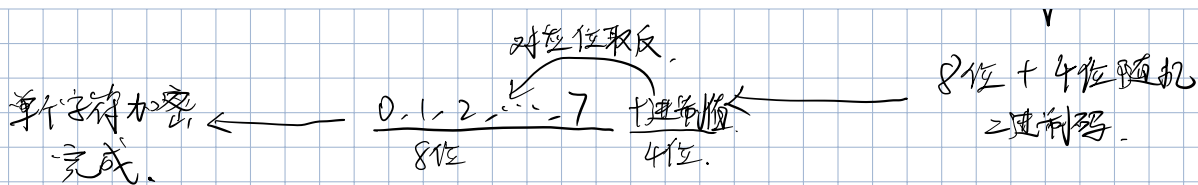
1. 基本流程



2. 加密, 解密方法介绍

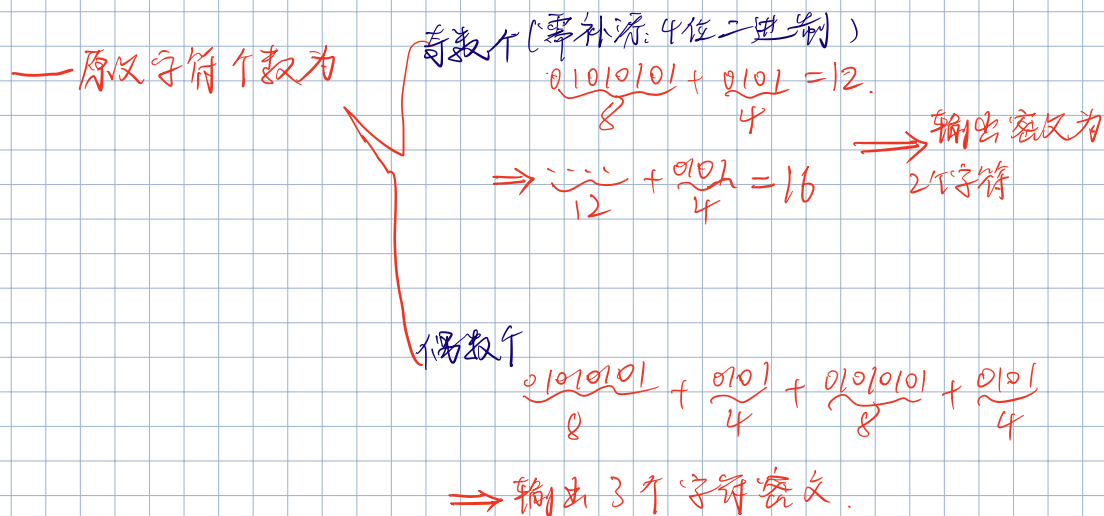
① 加密:





其中：4位二进制对应十进制取值范围为 $[0, 7]$ 。

—— ASCII $[0 \sim 127]$ 字符 —— \rightarrow 二进制为 7 位，首位为 0 位，不显示。



—— 使用 map 集合存储 随机 4 位二进制码 和 对应十进制值。

\Rightarrow `hashmap < 十进制, 二进制 >`。

$\Rightarrow [0 \sim 7] \rightarrow$ 二进制码会不满 4 位，需补 0。

② 解密：

string 密文 \rightarrow 二进制字符串 \rightarrow 每隔 12 位切割

解密完成
单字符

4位二进制对应十进制值
前8对应二进制位取反。

前8为取反的二进制字符串
后4为解密规则(前8位)

其中：—— 生成的密文中含有 0~9 数值中，生成二进制字符串也
不够8位。

—— 每12位对应一位原文字符。

3. 方法介绍 使用到的方法

① `char[] chars = str.toCharArray()`

② `String.valueOf(char)`

③ `StringBuilder.charAt(int)`

④ `StringBuilder.delete(int start, int end)`

⑤ `StringBuilder.insert(int num, String str)`

⑥ `StringBuilder.append(StringBuilder)`

★ ⑦ `Map.Entry<Object, Object>`

⑧ `StringBuilder.substring(int start, int end)`

⑨ `Integer.parseInt(String, 进制类型)`

⑩ `String.concat(String)`

⑪ `Integer.toString(int)` (int 3) \Rightarrow 11
二进制
(String "3") \Rightarrow 11 0011.