

```
1: //include <iostream>
2: //include <string>
3: #include "LFSR.hpp"
4: using namespace std;
5:
6: LFSR::LFSR(string seed, int tap) {
7:     for (unsigned int i = 0; i < seed.length(); i++)
8:         this->seed.push_back(seed[i]);
9:     //save_seed = seed;
10:    //since the tap is counted from right to left, must take total lengt
h
11:    //and subtract it from the input tap
12:    this->tap = seed.length() - tap - 1;
13: }
14:
15: int LFSR::step() {
16:     int first = seed.at(0),
17:         _tap = seed.at(tap),
18:         n_bit = first ^ _tap;
19:     seed.erase(seed.begin());
20:     seed.push_back(n_bit);
21:     return n_bit;
22: }
23:
24: int LFSR::generate(int k) {
25:     int val, output = 0;
26:     for (int i = 0; i < k; i++) {
27:         val = step();
28:         output = (output * 2) + val;
29:     }
30:     return output;
31: }
32:
33: ostream& operator<< (ostream &out, const LFSR &obj) {
34:     for (unsigned int i = 0; i < obj.seed.size(); i++) {
35:         out << obj.seed[i];
36:     }
37:
38:     return out;
39: }
40:
41: //LFSR::~~LFSR() {}
```