// Java program to generate n-bit Gray codes

import java.util.\*;

class GfG {

// This function generates all n bit Gray codes and prints the

// generated codes

static void generateGrayarr(int n)

{

// base case

if (n <= 0)

return;

// 'arr' will store all generated codes

ArrayList<String> arr = new ArrayList<String> ();

// start with one-bit pattern

arr.add("0");

arr.add("1");

// Every iteration of this loop generates 2\*i codes from previously

// generated i codes.

int i, j;

for (i = 2; i < (1<<n); i = i<<1)

{

// Enter the prviously generated codes again in arr[] in reverse

// order. Nor arr[] has double number of codes.

for (j = i-1 ; j >= 0 ; j--)

arr.add(arr.get(j));

// append 0 to the first half

for (j = 0 ; j < i ; j++)

arr.set(j, "0" + arr.get(j));

// append 1 to the second half

for (j = i ; j < 2\*i ; j++)

arr.set(j, "1" + arr.get(j));

}

// print contents of arr[]

for (i = 0 ; i < arr.size() ; i++ )

System.out.println(arr.get(i));

}

// Driver program to test above function

public static void main(String[] args)

{

generateGrayarr(3);

}

}