#include <iostream>

#include <stdio.h>

#include <stdint.h>

using namespace std;

void PrintBin(uint8\_t Val) {

int b;

for(b = 7; b >= 0; b--)

{

if (Val & (1 << b)){

putchar('1');

}

else{

putchar('0');

}

}

}

void first\_task (){

int number = 1;

bool left=true;

for(int i=0; i < 15; i++){

PrintBin(number);

printf("\n");

if (left){

number <<= 1;

if ((number >> 7) & 1u){

left = false;

}

}else{

number >>= 1;

if ((number >> 0) & 1u){

left = true;

}

}

}

}

void second\_task\_part\_one(int position=3, bool left=true, int number =1){

for(int i=0; i < 15; i++){

if ((number >> 3) & 1u){

PrintBin(number);

}

else{

number |= (1 << position); // set 1 in position 3

PrintBin(number);

number &= ~(1 << position); // del 1 from position 3

}

printf("\n");

if (left){

number <<= 1;

if ((number >> 7) & 1u){

left = false;

}

}else{

number >>= 1;

if ((number >> 0) & 1u){

left = true;

}

}

}

}

void second\_task\_part\_two(int position=0, bool left=true, int number =1){

for(int i=0; i < 20; i++){

PrintBin(number);

printf("\n");

if (left){

number <<= 1;

number |= (1 << position); // set 1 in position 0

if ((number >> 7) & 1u){

left = false;

}

}else{

number &= ~(1 << position); // del 1 from position 0

number <<= 1;

if (!((number >> 7) & 1u)){

left = true;

}

}

}

}

void second\_task\_part\_three(int position=0, bool left=true, uint8\_t number =1, int ending=7){

int \*ending\_positions = new int;

int len\_ending\_position = 0;

for(int i=0; i < 30; i++){

if (len\_ending\_position == 7){

PrintBin(255);

printf("\n");

continue;

}

PrintBin(number);

printf("\n");

number <<= 1;

if ((number >> ending) & 1u){

number |= (1 << position); // set 1 in position 0

ending\_positions[len\_ending\_position] = ending;

ending -= 1;

len\_ending\_position += 1;

}

for (int end\_position=0; end\_position < len\_ending\_position; end\_position++){

number |= (1 << ending\_positions[end\_position]); // set 1 in ending\_position

}

}

delete ending\_positions;

}

int main(){

int task\_numbers[3] = {5, 8, 9 };

// first\_task();

// second\_task\_part\_one();

// second\_task\_part\_two();

second\_task\_part\_three();

}