

```
final int BIN = 1;
int testBit = 0;
int randNum;
```

```
for(int i =0; i < 10; i++){
    //0~9가 나온다
    randNum = (int)(Math.random()*10);
    while ((testBit & (BIN<<randNum)) !=0){
        randNum = (int)(Math.random()*10);
    }
    System.out.printf("randNum = %d\n", randNum);
    testBit |= (BIN << randNum);
}
```

BIN이 1이고  $1 \ll \text{randNum}$  을 하면 randNum의 자릿수가 하나씩 증가된다.  
그상태로 `testBit &` 에 들어가면 비트연산자 `and` 에 의해 중복된 값이 나올경우 `while`이 계속 돌아가는것 같다.  
그 상태로 계속 프로그램을 돌리면 0~9까지 중복되지 않을때까지 계속 반복하다가 모든 수가 나오면 `while` 이 종료된다.

맨 아랫줄은 `testBit | (BIN << randNum)` 가 하는 역할은 잘 모르겠다.

Challenge 문제는 노력해봤지만 실패했습니다.  
하지만

```
int i = (int) (Math.random() * 5 + 5);
```

이런식으로 `math.random`을 설정한다면 5~9까지 수를 출력할 수 있다는 것을 알게되었습니다.

7~9를 출력하고 싶다면 이렇게 하면된다.

```
int i = (int) (Math.random() * 3 + 7);
```

하지만 두가지 조건이 중복되지 않게 하는 방법은 잘 모르겠다.