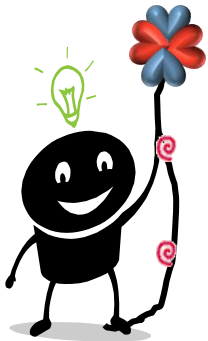


# 제 1과목 - 알고리즘

## <part2. 알고리즘 순서도작성>

### 10. 수학관련 알고리즘 - 보수계산



오늘의 핵심내용

1의 보수 변환 알고리즘 학습

2의 보수 변환 알고리즘 학습



보수(Complement)계산 알고리즘

알고있는내용

- ✓1의 보수 - 입력값을 반대 값으로 표시(1->0, 0->1)
- ✓2의 보수 - 1의 보수 계산결과 + 끝자리 1

10011

→

01100

1의 보수

0110

0

+

1

0110

1

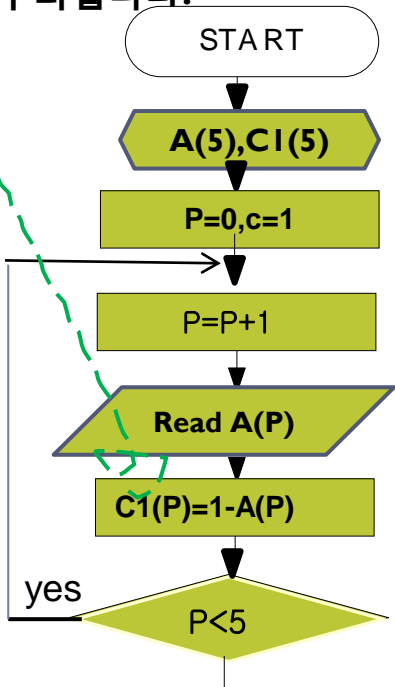
2의 보수

출제 포인트) 우리 시험에서는 배열에 입력된 2진수를 1의 보수 혹은 2의 보수로 계산하는 알고리즘이 출제되고 있습니다. ➔ 그것도 여러 번 출제됐습니다.

How to) 1의 보수



1'을 0으로 변환하기 위해 우리시험에서 요구한 알고리즘은 "1-해당 값" 입니다. 즉,  $1-1=0$  그래서, 0 이해되죠? 그럼 2번째 배열 값 '0'은  $1-0=1$  그래서, 1이 됩니다. 이런식으로 4자리 반복해서 작업수행하면 1의 보수는 마무리됩니다.



- 배열 A(5)- 2진수 5자리 기록된 공간
- 배열 C1(5) - 1의 보수 저장 공간
- P- 2진수 각 자릿수 값

How to) 2의 보수 → 올림수 발생시 어떻게 처리할까?

Step 1) 맨 마지막 자리에 1 더합니다. 그리고, 그 결과값을 확인합니다.

0

1

0

0

+

1

1

0

1

1

1

+

1

2

Step 2) 2진수이므로 2는 표시되면 안되죠? 따라서, 2로 현재 계산한 값을 나눠줍니다. 즉, 1/2 , 2/2 이렇게요... 이때 나머지 값이 바로 2의 보수 맨 마지막 자리에 들어가게 됩니다.

0

1

0

1

0

1

1

0

Step 3) 우리는 올림수 발생유무를 똑똑히 잘 알지만 프로그램은 바보라서 모릅니다. 따라서, 1의 보수 배열의 맨 마지막 값에 1을 곱합니다.

0

1

0

0

×

1

자리올림수

0

0

1

1

1

×

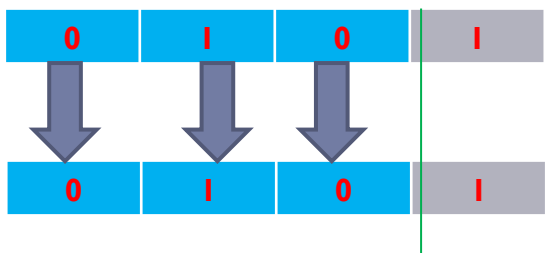
1

자리올림수

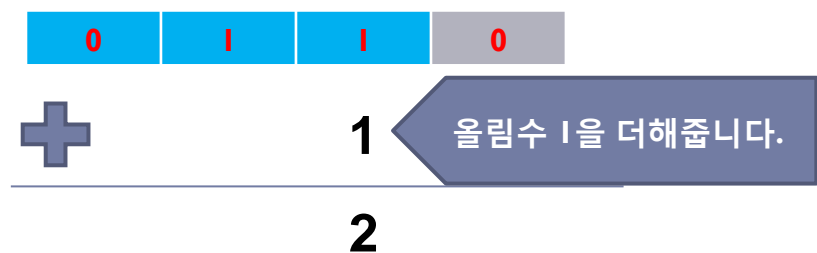
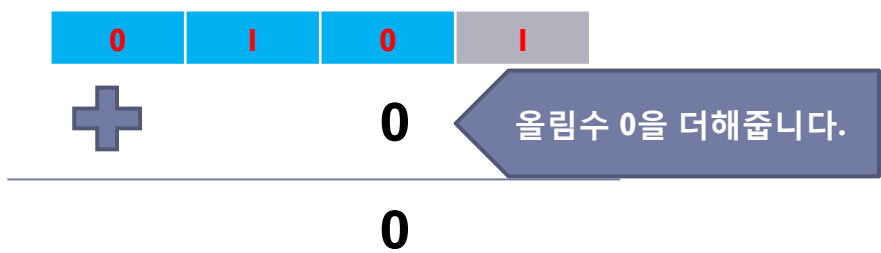
1



Step 4) 올림수가 0이 나오면 사실 앞의 값들은 그대로 기술해주면 작업이 끝납니다.  
=> 그런데, 시험에서는 이렇게 출제되면 쉬우니까 좀 반복을 시키고 있습니다. 아래 계속 보시죠.



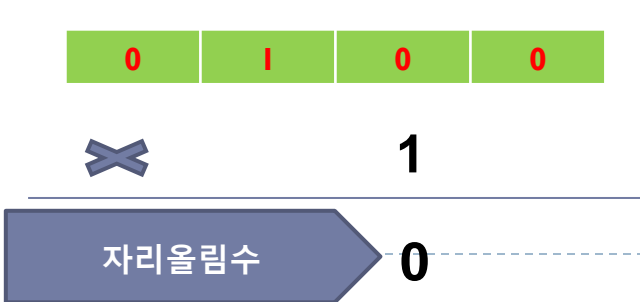
\* 우리 시험에서는 위 방법대신 아래와 같은 방법으로 해결을 했습니다.



\* 다시 2로 나누어 나머지 계산하여 1을 곱해줍니다.( step2-3을 반복실행합니다.)



\* 즉, 1의 보수의 세번째(맨 뒤 2번째) 값에 1을 곱하여 올림수 유무를 확인합니다.



\* 해당 올림수 값을 더해줍니다.

0

1

0

1

+

0

올림수 0을 더해줍니다.

1

0

1

0

0

+

1

올림수 1을 더해줍니다.

2

\* 다시 위의 계산 결과값을 2로 나누어 나머지 계산하여 2의 보수에 저장하고, 1의 보수 값에 대해 1을 곱해줍니다.

0

1

0

1

×

1

자리올림수 0

0

0

0

0

×

1

자리올림수 1

\* 즉, 1의 보수의 세번째(맨 뒤 2번째) 값에 1을 곱하여 올림수 유무를 확인합니다.



\* 해당 올림수 값을 더해줍니다.

0

1

0

1

+

0

올림수 0을 더해줍니다.

0

0

0

0

0

+

1

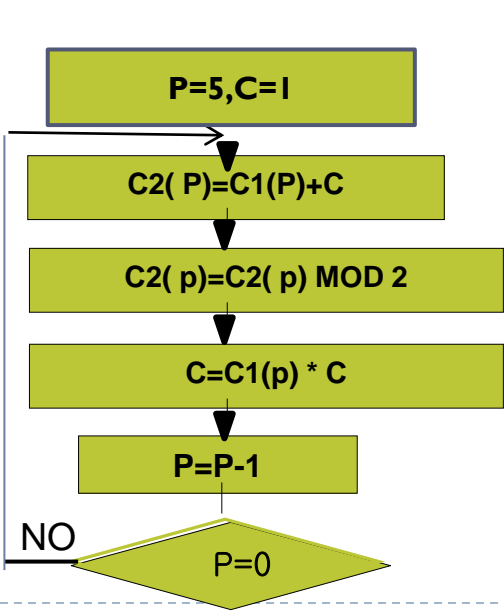
올림수 1을 더해줍니다.

1

\* 다시 위의 계산 결과값을 2로 나누어 나머지 계산하여 2의 보수에 저장해줍니다.

0	1	0	1
---	---	---	---

1	0	0	0
---	---	---	---



C1(5) -1의 보수 저장공간

C1(1)	C1(2)	C1(3)	C1(4)	C1(5)
0	1	0	1	1

\* MOD 연산자는 나머지 값을 계산, 예) 5 MOD 2 →1

C2(5) -2의 보수 저장공간

C2(1)	C2(2)	C2(3)	C2(4)	C2(5)

핵심곡!곡! 문제

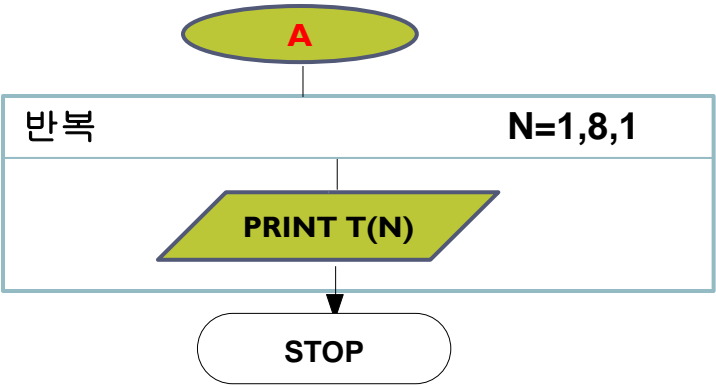
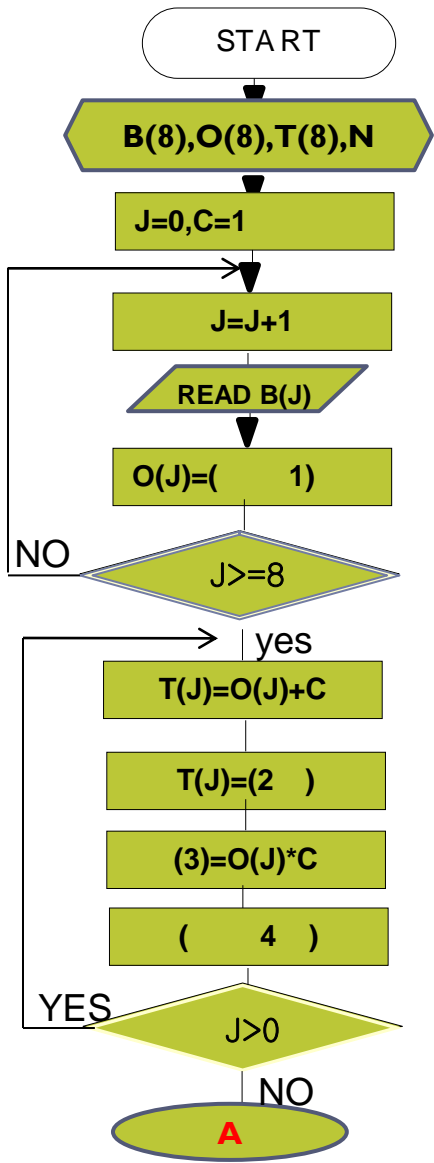
1. 크기가 8인 배열 B(8)에 입력되어 있는 2진수 8자리에 대하여 2의 보수를 구하여 출력하는 순서도 작성

처리조건

- 크기가 8인 배열 B(8)에 입력되어 있는 이진수 값의 범위는 00000001 ~ 11111111이며, 부호비트는 없는 것으로 간주한다.
- 각 배열의 첫 번째 요소 B(1), O(1), T(1)을 최상위 비트(MSB)로 간주한다.
- j,N -배열의 위치를 나타내는 인덱스
- o(8)-1의 보수가 저장될 배열, t(8) - 2의 보수가 저장될 배열
- c -자리 올림수가 저장될 변수







- (1) 답 -
- (2) 답 -
- (3) 답 -
- (4) 답 -

답 : \_\_\_\_\_

정답을 확인해요.

- 답 : (1) 답 - 1-B(J)  
(2) 답 - T(J) MOD 2  
(3) 답 -C  
(4) 답 -J=J-1

2진수 1100 1000 이 있을 때 이를 1의 보수 ,2의 보수로 변환하는 작업이에요.

B(1)	B(2)	B(3)	B(4)	B(5)	B(6)	B(7)	B(8)
1	1	0	0	1	0	0	0

1-B(1)    1-B(2)    .....    1의보수 결과 => 1-2진수 각 자릿수

O(1)	O(2)	O(3)	O(4)	O(5)	O(6)	O(7)	O(8)
0	0	1	1	0	1	1	1

+

1  
C의 역할 자리올림수

2진수 이므로 2가 나오면?  
0이 되어야겠죠? 따라서,  
2로 나눈 나머지값을 2의보수에  
저장하게 돼요.

2의 보수 계산방향

2의보수 결과

T(1)	T(2)	T(3)	T(4)	T(5)	T(6)	T(7)	T(8)
0	0	1	1	1	0	0	0

