

## Lab 3

두 명씩 짝을 지어 코딩 하는데, 한 학생은 개발자 역할을 다른 학생은 코치 역할을 수행합니다.

10분마다 역할을 바꿉니다.

각 문제를 명기한 시간 안에 완성하고 조교의 점점을 받고 다음 문제로 넘어갑니다.

각 문제별로 정해진 시간을 초과할 수 없습니다.

시간 내에 해결하지 못한 경우 다음 문제로 넘어가고,

풀지 못한 문제는 집에 가져가서 혼자서 풀도록 합니다.

해답은 추후에 제공합니다.

### 문제 1 (30분)

강의 시간에 공부한 `squeeze` 함수를 (`squeeze.c` 파일에서 제공) 다음 함수 프로토타입에 맞추어 문자열 `s2`에 있는 문자를 문자열 `s1`에서 모두 지우도록 프로그램을 수정하십시오.

```
void squeeze(char s1[], char s2[]);
```

다음을 실행하면

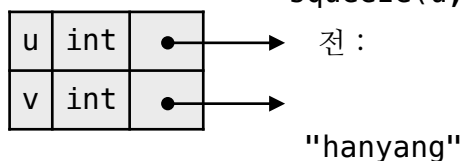
```
char u[10], v[5];
```

```
u = "hanyang";
```

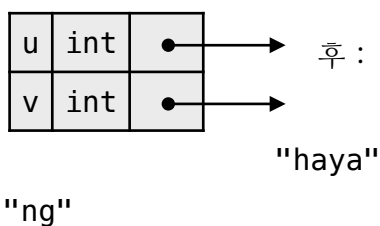
```
v = "ng";
```

```
squeeze(u,v);
```

호출



호출



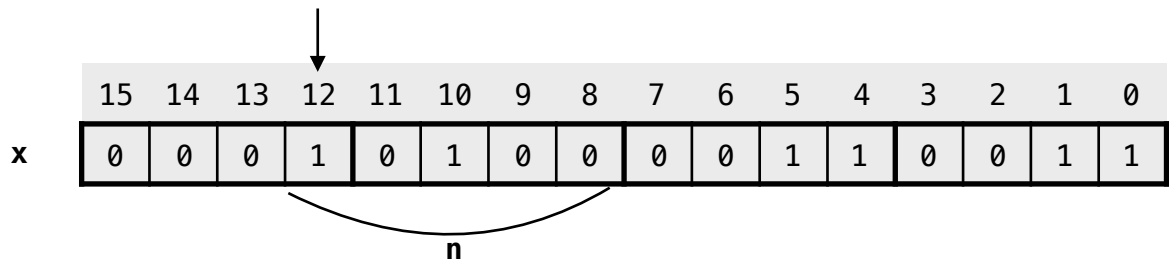
## 문제 2 (30분)

강의 시간에 공부한 `getbits` 함수의 짝이 되는 `setbits` 함수를 작성해보자. `setbits` 함수의 프로타입은 다음과 같은데,

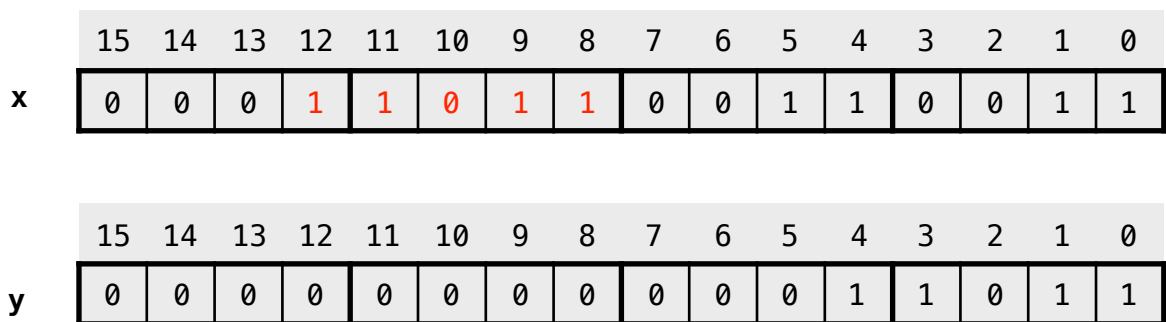
```
void setbits(unsigned x, int p, int n, unsigned y);
```

`x`의 `p` 위치에서 시작하여 연결된 `n`개의 비트를 `y`의 오른쪽 끝 `n`개 비트로 대치한다.

예를 들어, 다음과 같은 상황에 `p = 12, n = 5` 로 `setbits`를 실행하면 아래와 같이 된



실행 전 :



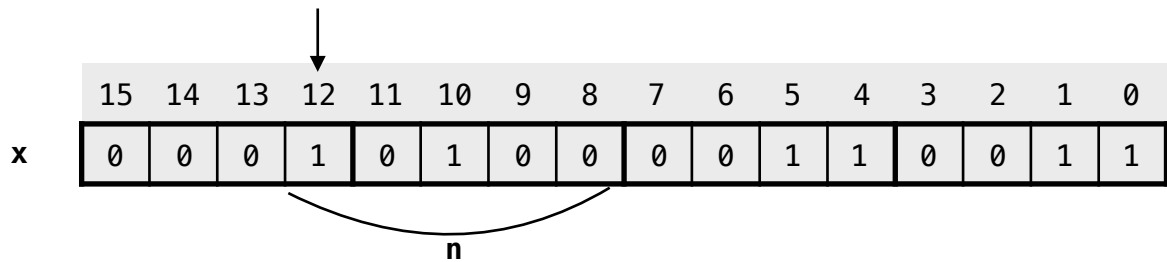
실행 후 :

**문제 3 (30분)**

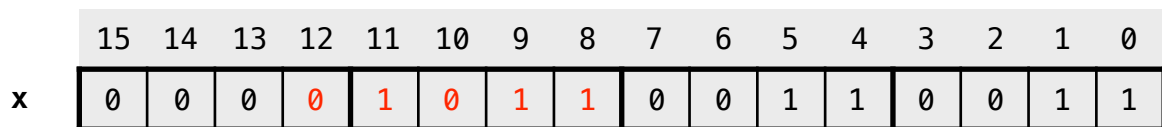
x의 p 위치에서 시작하여 연결된 n개의 비트를 0은 1로 1은 0으로 모두 뒤집고 나머지 부분은 그대로 두는 함수 invert를 작성하자. invert 함수의 프로토타입은 다음과 같다.

```
void invert(unsigned int x, int p, int n);
```

예를 들어, 다음과 같은 상황에 **p** 서  $p = 12, n = 5$  로 invert를 실행하면 아래와 같이 된다.



실행 전 :



실행 후 :

**문제 4 (30분)**

다음과 같은 프로토타입을 만족하는 함수를 작성하자.

```
unsigned circular_shift(unsigned x, int n);
```

이 함수는  $x$ 를  $n$  비트만큼 왼쪽으로 이동하는 함수인데, 이동할 때 상위비트(왼쪽끝)는 다시 하위비트

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
$x$	0	1	0	1	0	1	0	0	0	0	1	1	0	0	1	1

(오른쪽끝)로 들어가야 한다.

실행 전 :

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	1	0	1	0	0	0	0	1	1	0	0	1	1	0	1	0

`circular_shift(x,3)` 호출 후 내주는 수 :