

## PART 10-2. 배열을 처리하는 함수

### ◆ 연습문제 ◆

1. 다음과 같이 선언된 함수가 있을 때, 함수의 호출이 적합하지 않은 것은?

```
void func(int *p);
```

- ① `int ary[5];`  
    `func(ary);`
- ② `int ary[] = {1, 2, 3};`  
    `func(ary);`
- ③ `double ary[5];`  
    `func(ary);`
- ④ `int ary[5];`  
    `func(ary + 2);`

정답 : ③ `double ary[5];`  
    `func(ary);`

해설 : 없음

다음은 각 달의 일수를 저장한 배열입니다. `month[0]`은 1월이며 첨자가 순차적 증가할 때 월도 함께 증가합니다. (2번~4번)

```
int month[12] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
```

2. `month` 배열에서 총 일수가 31일인 달의 개수를 반환하는 함수를 만드세요.

정답

```
int count_31(int *mp) {
```

```

int cnt = 0;
int i;

for (i = 0; i < 12; i++) {
    if(mp[i] == 31) {
        cnt++;
    }
}

return cnt;
}

```

해설 : 없음

3. month 배열의 2월의 일수를 29일로 바꾸는 함수를 만드세요.

정답

```

void change_29(int *mp) {
    mp[1] = 29;
}

```

해설 : 없음

4. month 배열의 값을 한 줄에 5개씩 출력하는 함수를 만드세요.

정답

```

void print_month(int *mp) {
    int i;

    for (i = 0; i < 12; i++) {
        printf("%5d", mp[i]);
    }
}

```

```

        if ((i + 1) % 5 == 0) {
            printf("Wn");
        }
    }
}

```

해설 : 없음

5. 다음과 같이 초기화된 배열에서 각 값의 좌우 위치를 바꾸는 함수를 만듭니다. 배열 요소 수와 상관없이 바꿀 수 있도록 만드세요.

```
double ary[] = {1.5, 20.1, 16.4, 2.3, 3.5};
```

▶ 함수 호출 : reverse(ary, sizeof(ary) / sizeof(\*ary));

▶ 호출 후 바뀐 배열 요소의 값 : 3.5 2.3 16.4 20.1 1.5

정답

```

void reverse(double *pa, int size) {
    int i;
    double temp;

    for (i = 0; i < size / 2; i++) {
        temp = pa[i];
        pa[i] = pa[size - 1 - i];
        pa[size - 1 - i] = temp;
    }
}

```

해설 : 없음