

Java를 알고 C배우기

컴퓨터프로그래밍3

week 5-1 배열, 포인터-1차원배열

2022.1학기
충남대 조은선

배열의 필요성

```
printf("%d\n", rabbit1);  
printf("%d\n", rabbit2);  
printf("%d\n", rabbit3);  
...           // 100 마리 토끼라면 100개의 변수 선언?
```

```
printf("%d\n", rabbit[1]);  
printf("%d\n", rabbit[2]);  
printf("%d\n", rabbit[3]);
```

```
for (i = 0; i < 100; i++)  
    printf("%d\n", rabbit[i]);
```

▶ 인덱스를 써서 개별 변수를 지칭

배열 선언

`int rabbit[100];`

↑ ↑ ↑

요소 타입 배열 명 요소 개수

- ▶ 배열 선언(Element Type, Array Name, Number of Elements)
 - ▶ `int`는 배열의 타입이 아니라 배열을 구성하는 요소의 타입
- ▶ 배열의 모든 요소는 동종 자료형(Homogeneous Data Type)
 - ▶ `char char_set[26];`
 - ▶ `double math_score[100];`
- ▶ C 언어의 배열 인덱스는 0부터 시작
 - ▶ `int rabbit[100]`이면 요소 개수가 100개
 - ▶ 인덱스는 0부터 99
 - ▶ `rabbit[100]`은 존재하지 않음

배열 요소 접근

```
int rabbit [100];
```

```
int i;
```

```
rabbit[0] = 0;
```

```
rabbit[1] = 1;
```

```
i = 50;
```

```
rabbit[i] = i;
```

```
// rabbit[50] = 50;
```

```
rabbit[10] = rabbit[10] + 1;
```

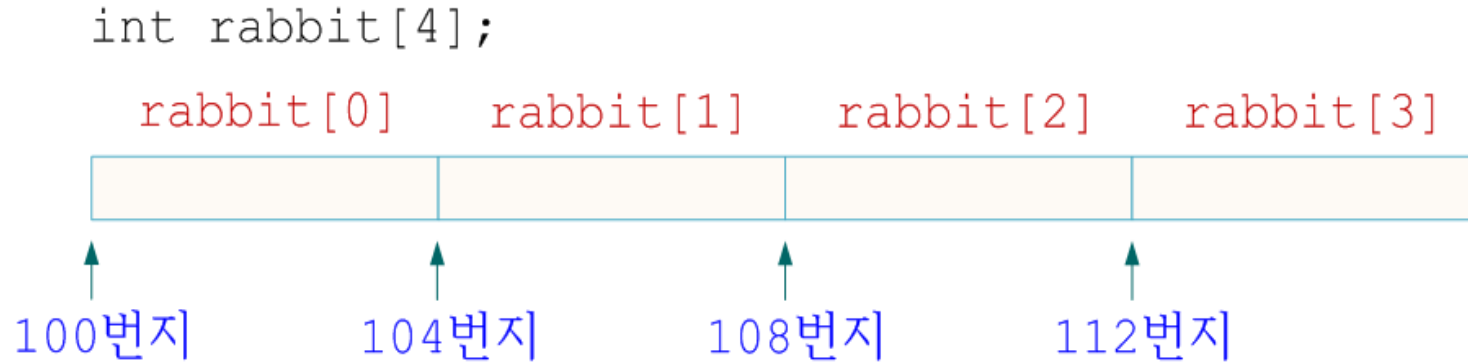
```
// 1증가
```

```
for (i=0; i < 100; i++)
```

```
    rabbit[i] = 0;
```

```
// rabbit[0] 부터 rabbit[99]까지 0으로 초기화
```

배열 요소의 주소



$$\&\text{rabbit}[i] = \&\text{rabbit}[0] + \text{sizeof}(\text{int}) * i$$

- ▶ 배열은 연속된 메모리 공간(Contiguous Memory Space)에 존재
 - ▶ 요소끼리 서로 붙어 있기 때문에 위 공식이 성립
 - ▶ cf. 만약 배열 인덱스를 1부터 시작하면 이는
 - ▶ $\&\text{rabbit}[i] = \&\text{rabbit}[1] + \text{sizeof}(\text{int}) * (i - 1)$ 로 바뀜
 - ▶ $(i - 1)$ 계산으로 인한 시간적 부담 초래

배열의 초기화

- ▶ 배열 선언과 동시에 초기화(cf. 정적 변수로 선언하면 자동으로 0)

```
int sales[4] = {5, 6, 7, 8};  
int sales[ ] = {5, 6, 7, 8};    // 알아서 크기 계산  
int sales[4] = {5, 6};         // 나머지는 0으로  
int sales[4] = {0};            // 모두 0으로
```

배열 요소의 개수

```
int days[ ] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31,
               30, 31};
int size_days = sizeof(days); // size of days[ ]
num_elts = sizeof(days) / sizeof(days[0]);

for (i = 0; i < num_elts; i++){
    printf("Month %d has %d days.\n", i + 1, days[i]);
}
```

- ▶ sizeof 연산자는 배열 전체나 배열 요소의 크기를 계산
 - ▶ 배열 전체의 크기 / 첫 요소의 크기 = 요소의 개수

Quiz

- ▶ `int rabbit[100];` 으로 선언된 배열에서 `rabbit[0]` 의 주소가 1000번지라면 `rabbit[42]` 의 주소는 무엇일까?