

K G 아 이 티 뱅 크

C 언 어

C L A N G U A G E

배열

배열 (Array)

❖ C언어에서 제공하는 값이 연속으로 배치된 자료구조

- 배열 변수는 배열의 시작위치와 배열의 크기를 담고 있는 변수
- 같은 자료형을 가진 값들의 모임이며, 다량의 데이터 저장에 유리
- 값은 요소라고 부르기도 하며 사용하기 위해 인덱스를 활용

❖ 코드. 선언과 초기화 예시.

```
int arr1[3]; // 배열의 선언. 상수만 가능
arr1[0]=1;   // 배열의 초기화
arr1[1]=2;   // 인덱스를 지정하여
arr1[2]=3;   // 지정한 공간에 저장
```

```
int arr2[3]={1,2,3}; // 선언 & 초기화1
// 고정된 크기에 맞춰 값을 초기화
```

```
int arr3[]={1,2,3}; // 선언 & 초기화2
// 저장하는 값의 갯수에 맞춰 선언
```

배열 (Array)

❖ 배열의 사용 : 인덱스를 이용하여 값을 저장 및 사용

❖ 코드1. 인덱스를 이용한 값의 저장 / 사용

```
arr1[0]=1;           // 배열에 값을 저장  
int result=arr1[0] * 2; // 배열의 사용
```

❖ 코드2. 인덱스를 이용한 값의 입력

```
printf("정수 입력 : "); // 인덱스 사용시  
scanf_s("%d", &arr[0]); // 변수와 동일취급
```

❖ 일반적으로는 반복문과 함께 연계하여 사용

❖ 코드3. 반복문과 연계한 사용

```
for (int i=1;i<=10;i++){  
    printf("%d번 값 입력 : ",i);  
    scanf_s("%d", &arr[i-1]);  
}
```

배열 (Array)

< 파일이름 : arrayEX1.c >

❖ 실습예제1. 아래의 조건을 달성하는 코드를 작성하세요.

조건

1. 크기가 3인 정수 배열, 실수 배열을 선언합니다.
2. 정수 배열에 0번 공간부터 10, 20, 30으로 초기화합니다.
3. 실수 배열에 0번 공간부터 1.1, 2.2, 3.3으로 초기화합니다.
4. 결과를 참고하여 연산결과를 출력합니다.

결과 (콜론 좌측은 저장된 값, 우측은 연산된 결과 출력)

(정수[0]) + (실수[1]) : (결과1)
(정수[1]) - (실수[2]) : (결과2)
(정수[2]) * (실수[0]) : (결과3)
(정수[0]) / (실수[1]) : (결과4)

배열 (Array)

< 파일이름 : ArrayEX2.c >

❖ 실습예제2. 아래의 조건을 달성하는 코드를 작성하세요.

조건

1. 크기가 5인 실수배열을 선언합니다.
2. 첫번째와 마지막은 값을 입력을 받아 저장합니다.
3. 나머지는 1.1, 2.2, 3.3으로 직접 초기화합니다.
4. 저장된 값의 출력은 반복문을 이용해 구성합니다.

결과(입력한 값이 5.5, 8.8 일 경우)

배열명[0] : 5.5
배열명[1] : 1.1
배열명[2] : 2.2
배열명[3] : 3.3
배열명[4] : 8.8

배열 (Array)

< 파일이름 : ArrayEX3.c >

❖ 실습예제3. 아래의 조건을 달성하는 코드를 작성하세요.

조건

1. 크기가 4인 정수배열을 선언합니다.
2. 반복문을 활용해 값을 입력을 받아 저장합니다.
3. 배열 내부의 값의 합을 구해 출력합니다.
4. 구한 합이 5의 배수여부를 확인해 출력합니다.

결과 (입력한 값이 1,2,3,4 일 경우)

배열에 저장된 값의 합

sum : 10

합은 5의 배수입니다.

배열 (Array)

< 파일이름 : ArrayEX4.c >

❖ 실습예제4. 아래의 조건을 달성하는 코드를 작성하세요.

조건

1. 국어, 영어, 수학 성적을 입력을 받아 배열에 저장합니다. 실수로 입력을 받습니다.
2. 입력을 받은 점수의 총합을 구해서 출력합니다.
3. 입력을 받은 점수의 평균을 구해서 출력합니다.
4. 평균이 80점 이상이면 ‘합격!’ 이라고 출력합니다.
5. 평균이 80점 이만이면 ‘불합격!’ 이라고 출력합니다.

결과

점수 총합 : 240점
점수 평균 : 80점
최종 평가 : 합격!

문자열

문자열 (String)

❖ 문자열 : 배열을 이용해 구현된 문자 자료형 값의 나열

- 기본적인 특징은 배열과 동일하며, 사용하는 방식도 동일

❖ 문자열은 아래와 같은 추가적인 특징을 가짐

- 단일문자상수로 표현되는 값은 한 개의 공간(1 Byte)만 사용
- 한글, 한자 등의 문자는 두 개의 공간(2 Bytes)을 사용
- 초기화를 동시에 진행할 경우, 문자열의 값을 복사하는 방식
- 문자열의 끝을 알리기 위해 'w0' 가 붙음(정수0과 동일)
 - **항상 저장할 문자열 + 1의 크기를 가져야 함**

❖ 코드

```
char word[8]="ABCDEFGG"
```

