

## 7 장 함수

---

### □ 개념 확인 학습

---

1. 함수란 무엇인가요?
2. 함수는 어떠한 형태를 가지고 있나요?
3. 표준 라이브러리 함수의 의미는 무엇인가요?
4. 표준 라이브러리 함수에서 문자 관련 함수에는 어떤 것이 있나요?
5. 표준 라이브러리 함수에서 문자열 관련 함수에는 어떤 것이 있나요?
6. 함수에서 return 문장이 하는 역할은 무엇인가요?
7. 값에 의한 호출(call by value)방식과 주소에 의한 호출(call by reference) 방식의 차이는 무엇인가요?
8. 지역 변수와 전역 변수는 어떻게 다른가요?

9. 정적(static) 변수의 특징은 무엇인가요?
10. 순환 호출(Recursive Call)이란 무엇인가요?
11. 정수형 배열을 함수의 매개변수로 전달하여 포인터로 받을 때 주의해야 할 점은 무엇인가요?
12. main() 함수가 매개변수를 가질 경우 사용형식은 어떻게 되나요?
13. 함수 포인터란 무엇인가요?

---

☐ 적용 확인 학습

---

1. 다음 서술 내용이 바르면 O, 그렇지 않으면 X를 표시 하세요.
  - A. 함수란 특정 작업이 반복되는 경우, 이 작업을 한 곳에 정의하고 필요할 때마다 호출해서 사용하는 프로그램내의 하나의 독립된 단위이다. ( )

- B. 함수에서 return 할 때에는 함수 자료형에 해당하는 값을 반환한다. ( )
- C. isalpha()는 알파벳인지 검사하는 함수이다. ( )
- D. strncmp()는 두 문자열을 대소문자를 구별하여 사전적으로 비교한다. ( )
- E. return은 값을 반환하는 문장이므로 식이 들어갈 수 없다. ( )
- F. 매개 변수 전달방식에서 주소에 의한 호출(call by reference) 방식은 포인터를 사용하여 작성할 수 있다. ( )
- G. 동일한 이름의 전역 변수와 지역 변수가 선언되어 사용되고 있다면, 현재 처리중인 명령문의 영역(함수 또는 블록)에서 선언된 변수를 우선적으로 처리한다. ( )
- H. 정적(static) 변수는 함수가 종료되어도 프로그램이 종료되기 전까지 값은 그대로 유지한다. ( )
- I. 순환 함수의 호출 후에 있는 명령문들은 스택에 push되었다가 프로그램이 종료하기 전 pop되어 실행된다. ( )

2. 다음 설명에 대한 함수의 함수헤더를 작성하세요.

- A. 함수 func()는 첫 번째 매개변수는 int이고 두 번째 매개변수는 double이며 반환 값은 없다.
- B. 함수 func()는 정수 매개변수 3개의 최댓값을 반환한다.

- C. 함수 func()는 매개변수는 문자열 배열 세 개이다. 이 함수에서는 전달 받은 첫 번째와 두 번째 문자열 배열을 연결하여 세 번째 문자열 배열에 저장한다.
- D. 함수 func()의 매개변수는 문자열 배열과 문자이며, 반환 값은 문자열 배열에서 전달 받은 문자가 있는 위치이다.
- E. 함수 func()는 첫 번째 매개변수는 실수형 배열이고 두 번째 매개변수는 정수형 배열이다. 이 함수에서는 전달 받은 실수형 배열의 절대값을 정수형 배열에 차례로 저장한다.
- F. 함수 func()는 문자열을 2개 포함하는 2차원 문자열 배열을 받아들여 첫 번째와 두 번째 문자열이 같으면 0, 그렇지 않으면 -1을 반환한다.

3. 다음 설명에 대한 변수 선언 문장을 작성하세요.

- A. 함수 원형이 void func(int, int)인 함수 포인터 pf
- B. 함수 원형이 int \*func(double, double)인 함수 포인터 addf
- C. 함수 원형이 double func(double \*, double)인 함수가 세 개인 함수 포인터 배열 aryfp

4. 오류를 찾아 수정하세요.

- A. 표에서 “함수의 호출 예” 부분.

함수의 원형 예	함수의 호출 예
void func(int *n);	int num=2; func(num);
void func(char *s);	char ch = "a" ; func(ch);
void func(int n[], int size); void func(int *n, int size);	int nums[5] = {5, 4, 3, 2, 1}; func(&nums, sizeof(nums[1])/sizeof(int));
void func(char s[]); void func(char *s);	char str= "abc" ; func(str);
void func(int (*n)[5], int size);	int nums[2][5] = {5, 4, 3, 2, 1, 1, 2, 3, 4, 5}; func(nums, sizeof(num[0])/sizeof(int));
void func(char (*s)[10], int size);	char strs[2][10] = {"apple", "banana"}; func(strs, sizeof(str)/sizeof(strs[1]));

B.

```

void mycmp(char *sa, char *sb) {
    return;
}

int main() {
    char stra[] = "apple";
    char strb[] = "apple";
    mycmp(&stra, &strb);
}

```

C.

```
int main(int *argc, char argv[]) {
    printf("argc = %d\n", argc);
    for (int i = 0; i < argc; i++)
        printf("argv[%d] = %s\n", i, argv[i]);

    return 0;
}
```

D.

```
void add(int *, int, int);
void sub(int *, int, int);
void (*pfarr[2])(int *, int, int) = { add[0], sub[1] };
```

---

### ☐ 응용 프로그래밍

---

1. main() 함수의 매개변수로 명령어 라인에서 받은 모든 문자열을 출력하는 프로그램을 작성하세요.

```
C:> ctest 1234 abcd 567
argc=4
argv[0] = ctest.exe
argv[1] = 1234
argv[2] = abcd
argv[3] = 567
```

```
C:> ctest "1234 abcd 567"
argc=2
argv[0] = ctest.exe
argv[1] = 1234 abcd 567
```

2. main() 함수의 매개변수로 명령어 라인에서 한 줄의 문자열을 입력 받아, 그 문자열에서 숫자만을 찾아내어 출력하는 프로그램을 작성하세요.

```
C:> ctest 1234abcd567efg
입력한 문자열에서 숫자는 -> 1234567
```

3. main() 함수의 매개변수로 명령어 라인에서 다음과 같이 입력 받아, 사칙 연산을 수행하는 프로그램을 작성하세요.

```
C:> ctest 20 + 30
-> 20 + 30 = 50

C:> ctest 20 * 30
-> 20 * 30 = 600
```

4. main() 함수의 매개변수로 명령어 라인에서 다음과 같이 입력 받아, 그 값을 표현하는 문자열로 출력하는 프로그램을 작성하세요. 10000보다 작은 정수로 입력 받으며 출력은 아래와 같은 방법으로 합니다.

```
C:> ctest 3496
입력한 정수는 [삼천 사백 구십 육]입니다.
```

5. 프로그램 시작 후 정수 세 개를 입력 받아 num 배열에 저장한 후 max()에 전달합니다. max()는 전달 받은 배열에서 최댓값을 반환(max\_value) 합니다. 이 내용을 수행하는 프로그램을 작성하세요.

```
정수입력 : 22
정수입력 : 55
정수입력 : 33
max()에서 전달 받은 최댓값 = 55
```

6. 정수 세 개를 main() 함수의 매개변수로 명령어 라인에서 입력 받아 num 배열에 저장한 후 max()에 전달합니다. max()는 전달 받은 배열에서 최대값(max\_value)을 반환합니다. 이 내용을 수행하는 프로그램을 작성하세요.

```
C:> ctest 22 55 33
max()에서 전달 받은 최대값 = 55
```

7. 프로그램 시작 후 문자열 두 개(stra, strb)를 입력 받아 저장한 후 입력받은 문자열 두 개와 결과가 저장될 문자열 한 개(plus)를 mystrcat()에 전달합니다. mystrcat()는 전달 받은 두 개의 문자열을 붙여 결과가 저장될 문자열 배열에 저장합니다. plus 문자열은 함수를 호출한 곳에서 출력합니다.

```
문자열 입력 : coffee
문자열 입력 : apple
plus = coffeeapple
```

8. main() 함수의 매개변수로 명령어 라인에서 문자열 두 개(stra, strb)를 입력 받아 저장한 후, 입력받은 문자열 두 개와 결과가 저장될 문자열 한 개(plus)를 mystrcat()에 전달합니다. mystrcat()는 전달 받은 두 개의 문자열을 붙여 결과가 저장될 문자열 배열에 저장합니다. plus 문자열은 함수를 호출한 곳에서 출력합니다.

```
C:> ctest coffee apple
plus = coffeeapple
```

9. myabs()는 실수 하나를 받아들여 절대값을 반환합니다. 임의의 수를 입력 받아 myabs()에 전달한 후 호출 결과를 출력하세요.



10. `isprime()`는 양의 정수 하나를 받아들여 소수인지 판별합니다. 임의의 수를 입력 받아 소수임을 확인하는 프로그램을 작성하세요. `isprime()`은 호출 결과로 소수일 경우 참인 1을 그렇지 않을 경우 0을 반환합니다.
11. 문자열을 입력 받아 널을 제외한 문자열의 길이를 반환하는 프로그램을 작성하세요. (`int mystrlen(char *)`)
12. 문자열과 문자를 한 개씩 입력 받아 문자열에서 문자의 개수가 몇 개인지 반환하는 프로그램을 작성하세요. (`int mystrcnt(char *, char)`)
13. 배열 `source`에 소문자 문자열을 입력 받아, `source` 배열의 내용을 대문자로 변환하여 배열 `target`에 저장하는 함수 `mystrupper()`를 포인터를 이용하여 작성하세요. `mystrupper()`에는 배열 `source`와 `target`을 전달하고 함수 수행 후 호출한 곳에서 결과를 출력합니다.
14. 문자열 두 개를 입력 받아 2차원 문자열 배열에 저장한 후, `strcmp2dim()`에 전달합니다. `strcmp2dim()`의 호출 결과로 첫 번째와 두 번째 문자열이 같으면 0 아니면 -1을 반환합니다.

15. 정수 입력은 메인함수의 매개변수로 5개를 입력 받습니다. mysort()는 입력 받은 정수 배열을 작은 것부터 큰 것의 순서로 정렬하는 함수입니다. mysort()의 매개변수는 두 개로 하나는 입력 받은 배열, 또 하나는 정렬된 배열입니다. 결과는 mysort() 함수를 호출한 곳에서 출력합니다.

```
C:> ctest 77 33 1 55 6
ary =    77  33   1  55   6
sortary =    1   6  33  55  77
```