

## 14 주차 Report

문제 1) 영문 문자열 안에 포함된 영단어의 개수를 계산하여 화면에 출력하는 프로그램을 작성하여 보자.

Hint : strtok() 함수를 사용하여 보자, 첫 번째 호출은 strtok(s, " ") 이고 두 번째 호출부터는 strtok(NULL, " ")와 같이 호출한다. 한 줄의 문자열을 gets\_s()를 사용하여야 한다.

- 문제 분석 & 동작 설명

힌트에 제시된 것 처럼 strtok() 함수를 사용하여 각 단어들을 분리하도록 하자. 입력받은 문자열을 저장할 문자열, 그 문자열의 성분의 수를 출력할 변수, 토큰 포인터를 선언하고 gets\_s()를 통해 문자열을 입력 받도록 하자. GCC 에서는 gets\_s() 함수는 없으므로 gets()로 입력받자. 입력받은 문자열을 strtok(s, " ")를 통해 토큰을 이용하여 단어 단위로 분리하도록 하자. 그리고 입력 받은 문자열의 널문자, 즉 문자열 끝까지 반복하도록 하고, 반복할 때마다 문자열의 성분의 수를 출력할 변수에 저장된 수를 1 씩 늘려주고 strtok(NULL, " ")를 통해 토큰을 얻도록 하자. 반복이 끝난 후, 최종적으로 끝나게 된 문자열의 성분의 수를 출력해주고 프로그램을 종료하자.

- 프로그램 소스 (주석 포함)

```

main.c [Week14 Report 1] - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Debug
<global>
main.c x
1 // 단어의 수 판별기
2 #include <stdio.h> // STanDard Input Output 함수 헤더파일 전처리기
3
4 int main(void) // main function
5 {
6     char *token; // 문자열 토큰 분리 포인터 token 선언
7     int output = 0; // 입력받은 문자열의 단어의 수를 출력할 정수형 변수를 0으로 초기화
8
9     char input[100]; // 입력받은 문자열을 저장할 size가 100인 문자열 선언
10    printf("문자열을 입력하십시오 : "); // 문자열을 입력해라는 msg 출력
11    gets(input); // 문자열 입력 받음 (gcc 컴파일러라 gets가 없음)
12
13    token = strtok(input, " "); // 문자열 토큰을 분리해주는 strtok 함수 사용하여 문자열을 전달하고 토큰을 얻음
14    while(token != NULL) // token 안에서 NULL 문자가 나오기 전까지 아래의 문장을 반복
15    {
16        output++; // 단어 수를 출력할 변수를 1씩 증가
17        token = strtok(NULL, " "); // 토큰을 얻음
18    }
19
20    printf("단어의 수는 %d 입니다.", output); // 바보문 다 돌고 나오 output 출력
21
22    return 0; // main function이 0으로 반환
23 }
24

```

```

D:\학교\대학교\충북대학교\1학년\기초컴퓨터프로그래밍 10분반\14주차\
문자열을 입력하십시오 : i am a boy
단어의 수는 4 입니다.
Process returned 0 (0x0)   execution time : 3.496 s
Press any key to continue.

```

문제 2) 다음과 같이 연산의 이름을 문자열로 받아서 해당 연산을 실행하는 프로그램을 작성하다.  
연산을 나타내는 문자열은 "add", "sub", "mul", "div"로 하라.

Hint : 문자열들이 서로 일치하는지를 비교할 때는 strcmp()를 사용한다.

다 알려주는 Hint :

```
printf("연산을 입력하시오 : ");  
gets(s);  
token = strtok(s, " ");  
strcpy(op, token);  
token = strtok(NULL, " ");  
x = atoi(token);  
token = strtok(NULL, " ");  
y = atoi(token);
```

- 문제 분석 & 동작 설명

입력 받는 연산자, 두 수를 저장할 문자열과 변수 2 개와 연산한 값을 출력할 변수 1 개를 각각 선언해주도록 하자. 입력받는 연산자를 저장할 문자열의 크기는 3 자리로 하는 것이 충분하겠다. 사용자들에게 앞서 언급한 변수들에 들어갈 값들을 입력 받자. 조건문과 Hint 의 strcmp() 함수를 이용하여 저장된 문자열이 각각 "add", "sub", "mul", "div"가 맞는지 판별하도록 하고, 각 상황에 맞는 연산을 할 수 있도록 조건문을 구성해주면 되겠다. 다 알려주는 Hint 에는 토큰의 분리와 형 변환 함수를 이용하여 진행했는데, 그것 보다는 앞서 언급한 조건문을 사용하는 것이 훨씬 간단할 것으로 생각된다. 연산한 후에는 값을 출력해주고 프로그램을 종료하면 되겠다.

- 프로그램 소스 (주석 포함)

```

main.c [Week14 Report 2] - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Debug
<global>
main.c x
1 #include <stdio.h> // STanDard Input Output 함수 헤더파일 전처리기
2 #define SIZE 3 // 배열의 크기를 3으로 정의
3
4 int main(void) // main function
5 {
6     char yeonsan[SIZE]; // 입력받은 연산자를 저장할 문자열 yeonsan 선언
7     int a, b, output; // 연산할 두 수와 출력 값을 저장할 정수형 변수 a, b, output 선언
8
9     printf("연산을 입력하십시오: "); // 사용자에게 연산을 입력해라는 msg 출력
10    scanf("%s %d %d", &yeonsan, &a, &b); // 연산자, a, b 순으로 입력받아 문자열과 각 변수에 저장
11
12    // 문자열들이 서로 일치하는지를 비교할 때는 strcmp()를 사용한다.
13    if(strcmp(yeonsan, "add") == 0) // yeonsan에 저장된 문자열이 add가 참이면
14        output = a + b; // 두 수를 더한다
15
16    else if(strcmp(yeonsan, "sub") == 0) // yeonsan에 저장된 문자열이 sub가 참이면
17        output = a - b; // 두 수를 뺀다
18
19    else if(strcmp(yeonsan, "mul") == 0) // yeonsan에 저장된 문자열이 mul가 참이면
20        output = a * b; // 두 수를 곱한다
21
22    else if(strcmp(yeonsan, "div") == 0) // yeonsan에 저장된 문자열이 div가 참이면
23        output = a / b; // 두 수를 나눈다
24
25    printf("연산의 결과: %d", output); // 연산 한 값을 출력
26
27
28    return 0; // main function이 0으로 리턴
29 }
30
31

```

```

D:\학교\대학교\충북대학교\1학년\기초컴퓨터프로그래밍 10분반\14주차\과제
연산을 입력하십시오: add 3 5
연산의 결과: 8
Process returned 0 (0x0)   execution time : 10.911 s
Press any key to continue.

```

문제 3)

3-1. 다음과 같은 데이터들을 가지는 구조체를 정의하고 c1 이라는 이름의 구조체 변수를 정의하여 보라.

```
struct customer {  
    char name[];  
    int address;  
    long mileage;  
};  
  
struct customer c1;
```

3-2. 다음의 설명에 부합하는 구조체를 정의하여 보라.

(a) char title[30]과 int pub\_date, pages, price 를 포함하는 구조체 book

```
struct book{  
    char title[30];  
    int pub_date;  
    int pages;  
    int price;  
};
```

(b) char name[30], int age, double height 를 포함하는 구조체 friend

```
struct friend{  
    char name[30];  
    int age;  
    double height;  
};
```

(c) 이름, 수량, 가격으로 부품 재고를 표현하는 구조체 part 정의

```
struct part{  
    char name[20];  
    int quantity;  
    int price;  
};
```

#### 문제 4)

원의 중심을 나타내는데 point 구조체를 사용할 수 있다. 원을 나타내는 circle 구조체를 정의하라. 이 circle 구조체를 받아서 다음과 같은 기능을 하는 함수를 작성하고 테스트하라.

```
struct point{
    int x, y;
};

struct circle{
    struct point center; // 원의 중심
    double radius;       // 원의 반지름
};
```

- (a) 원의 면적을 계산하는 함수 double area(struct circle c)
- (b) 원의 둘레를 계산하는 함수 double perimeter(struct circle c)

Hint : 구조체를 함수로 넘기면 복사본이 전달된다.

- 문제 분석 & 동작 설명

문제에 제시된 것처럼 원의 중심을 구성하는 변수 x, y 를 한번에 묶는 구조체와, 원의 수치들을 구성하는, 원의 중심의 구조체와 원의 반지름의 변수 radius 를 한번에 묶는 구조체를 이용하도록 하자. 메인 함수에서 구조체를 초기화 하고, 원의 면적과 둘레를 각각 계산하는 함수를 각각 만들어 연산을 한 후 문제의 예시대로 원의 중심점, 원의 면적, 원의 둘레를 각각 출력하도록 하면 되겠다. 주의해야 할 점은 구조체를 매개변수로 받을 때, 구조체 내의 어느 값이 아니라 구조체 자체가 카피된다는 것을 잊지 말아야 할 것이다.

- 프로그램 소스 (주석 포함)

```

1  #include <stdio.h> // STanDard Input Output 함수 헤더파일 전처리기
2  #define PI 3.14 // 원주율을 3.14로 정의
3
4  struct point { // 중심점의 x와 y좌표를 저장하는 점수형 변수를 묶은 구조체 point
5      int x, y; // 좌표평면상 중심점
6  };
7
8  struct circle { // 원의 성분들을 저장하는 변수들을 묶은 구조체 circle
9      struct point center; // 원의 중심
10     double radius; // 원의 반지름
11 };
12
13 double space(struct circle c1){ //원의 면적을 계산하는 사용자 지정함수
14     double area; // 계산한 면적을 저장할 실수형 변수 area 선언
15     area = PI * c1.radius * c1.radius; // 면적 계산 (c1.radius = 10)
16
17     return area; // 계산한 면적을 매인함수로 리턴
18 }
19
20 double perimeter(struct circle c1){ // 원의 둘레를 계산하는 사용자 지정함수
21     double round; // 계산한 둘레를 저장할 실수형 변수 round 선언
22     round = 2 * PI * c1.radius; // 둘레 계산
23
24     return round; // 계산한 둘레를 매인함수로 리턴
25 }
26
27 int main(void) // main function
28 {
29     struct circle c1 = {{0, 0}, 10}; // 구조체 c1 의 값을 초기화
30
31     printf("원의 중심 좌표 : (%d, %d)\n", c1.center.x, c1.center.y); // 구조체 c1의 중심점 구조체의 x와 y 좌표 각각 출력
32     printf("원의 반지름 : %0.0f\n", c1.radius); // 구조체 c1의 반지름 radius 출력
33     printf("원의 면적 = %f, 원의 둘레 = %f", space(c1), perimeter(c1)); // 구조체 c1을 매개변수 각각에 통째로 보사, 연산 한 값을 각각 출력
34
35     return 0; // 매인함수가 0으로 가버려요~
36 }
37

```

```

"D:\학교\대학교\충북대학교\1학년\기초컴퓨터프로그래밍 10분반\14주차\
원의 중심 좌표 : (0, 0)
원의 반지름 : 10
원의 면적 = 314.000000, 원의 둘레 = 62.800000
Process returned 0 (0x0)   execution time : 4.010 s
Press any key to continue.

```