C언어-함수

1. 함수란?
- 특정작업을 수행하는 명령어들의 모임의 이름
- 입력 받아서 특정한 작업을 수행해 결과 반환

2. 함수의 종류?
- 라이브러리함수 : 컴파일러에서 지원되는 함수 →
  - 표준입출력
  - 수학 연산
  - 문자열처리
  - 시간처리
  - 오류처리
  - 데이터 검색과 정렬
- 사용자정의함수 : 개발자가 직접 만들어서 사용하는 함수

3. 함수의 정의

```
void print_star()      ← 함수헤더 (반환형: void / 함수이름: print_star / 매개변수(현재x))
{                      ← 함수의 몸체
    for(int i=0; i<30; i++)
        printf("*");
}
```

4. 함수의 이름?
- 식별자에 대한 규칙을 따른 아무 이름. ⇒ 되도록이면 함수의 기능을 암시하는 이름붙여.

5. 함수의 몸체
- 함수가 수행하는 작업에 필요한 문장들
- 문장의 제한 X, 변수의 정의 O, 모든 제어구조의 사용 O, 다른함수의 호출 O

6. 함수의 호출
- 함수의 이름을 적으면 호출됨.
- 함수 호출시, 현재 실행하고 있는 코드 잠시 중단
  & 호출된 함수로 이동해 함수 몸체 안의 문장들 순차적으로 진행
  & 호출된 함수의 실행이 끝나면 호출한 위치로 다시 돌아가 잠시 중단되었던 코드 실행 재개

7. 반환형이란?
- 함수가 처리를 종료한 후에 호출한 곳으로 반환하는 데이터유형
- 반환하는 값이 없으면 void 사용

8. 함수의 예제

```
#include <stdio.h>
void print_stars()                 ← 함수정의
{
    for(int i=0; i<30; i++)
        printf("*");
}
int main(void)
{
    print_stars();                 ← 함수호출
    printf("\n Hello World! \n");
    print_stars();
    printf("\n");
    return 0;
}
```

9. 인수와 매개변수
- 함수 호출시 데이터를 주고 받기 위해 필요
- 인수(Argument) : 호출 프로그램에 의해 함수에 실제로 전달하는 값
- 매개변수(parameter) : 이 값을 전달받는 변수 ← 전달
- 함수 정의할 때, 각 매개변수에 자료형과 이름을 지정해야함.
- 주의할점 : (매개변수의 개수) = (인수의 개수) & (매개변수의 자료형) = (인수의 자료형)

```
value = max(10,20);   ⇒
```

```
int max(int x, int y)
{
    if(x>y)
        return x;
    else
        return y;
}
```

10. 반환값(return value)

- 함수가 호출한 곳으로 반환하는 작업의 결과값
- 인수는 여러 개가 될 수 있지만, 반환값은 하나만 가능
- 함수 사용시 꼭 반환값의 형을 명시할것.
- 반환값이 없다면 return;으로 마무리

11. 함수 호출예시

```
#include <stdio.h>
int max(int x, int y)      ⇒ 함수정의
    (↳함수이름)  (↳매개변수)
{
  if (x>y)
      return x;
  else
      return y;
}
int main(void)
{
  int x,y;
  printf("정수를 입력하시오: ");
  scanf("%d", &x);
  printf("정수를 입력하시오: ");
  scanf("%d", &y);

  int larger;
  larger = max(x,y);      ⇒ 함수호출
  printf("더 큰 값은 %d.\n", larger);
  return 0;
}
```

- 함수를 미리 정의하고, 호출해야 정상작동
  But. 나중에 정의하고 싶다면, 함수의 원형을 정의할것.

예시

```
#include <stdio.h>
double c_to_f(double c_temp); //함수원형정의

int main(void)
{
  printf("섭씨 %lf도 = 화씨 %lf \n", 36.0, c_to_f(36.0));
                                           (↳함수호출)
  return 0;
}

double c_to_f(double c_temp);      ⇒ 함수정의
{
  return 9.0 / 5.0 * c_temp + 32;
}
```

12. 난수함수(random number)

- 규칙성 없이 임의로 생성되는 수
- rand(): 의사난수(pseudo random number) 생성하는 함수
  ↳ 초기값에 따라서 나오는 순서가 어느정도 결정되어있는 난수
  : stdlib.h에 정의되어 있음

13. 변수의속성

범위(SCOPE) : 변수가 어떤 범위에서 사용이 가능한가
(=가시성(visibility))

- 지역변수 : 함수 또는 블록 안에서 정의되는 변수
  : 해당 블록이나 함수 안에서만 사용이 가능함.
  : 변수가 선언된 블록이 시작할 때 시스템스택(stack)이라 불리는 메모리공간에 만들어지며 동시에 초기화됨.
  : 함수의 헤더부분에 정의되어 있는 매개변수도 일종의 지역변수
- 전역변수 : 함수의 외부에서 선언되는 변수
  : 소스 파일의 어느 곳에서도 사용이 가능함.
  : 프로그래머가 전역변수를 초기화 안하면 컴파일러에 의해 '0'으로 초기화됨
  : 전역변수들로 인해 코드가 꼬이는 현상 : 스파게티 코드

⇒ (지역변수의 이름) = (전역변수의 이름)
"지역변수 우선!!"

생존시간(lifetime) : 변수가 메모리상에 얼마나 오랫동안 존재하는지
↳ 변수에게 할당된 기억공간이 얼마나 유지되느냐.