# C Camp

Software Department



#### Contents



C 언어란?

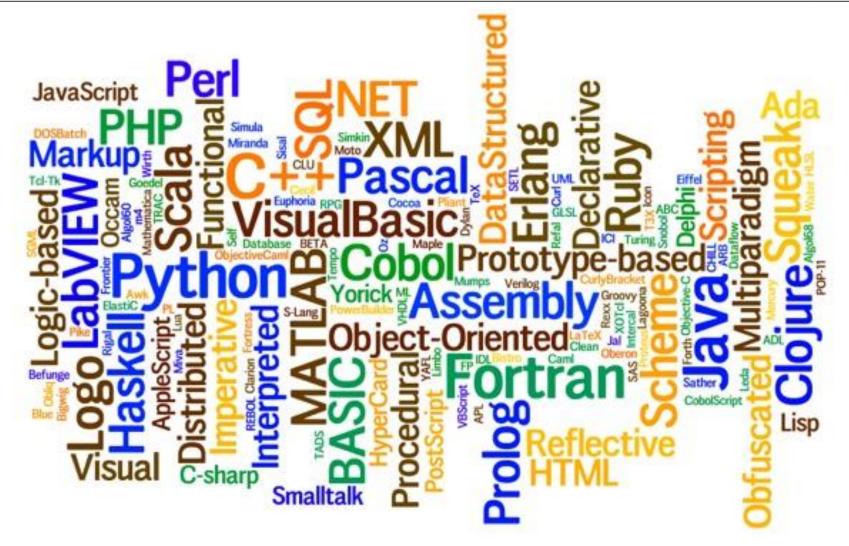
Compiler란?

변수란?

printf 함수와 scanf 함수

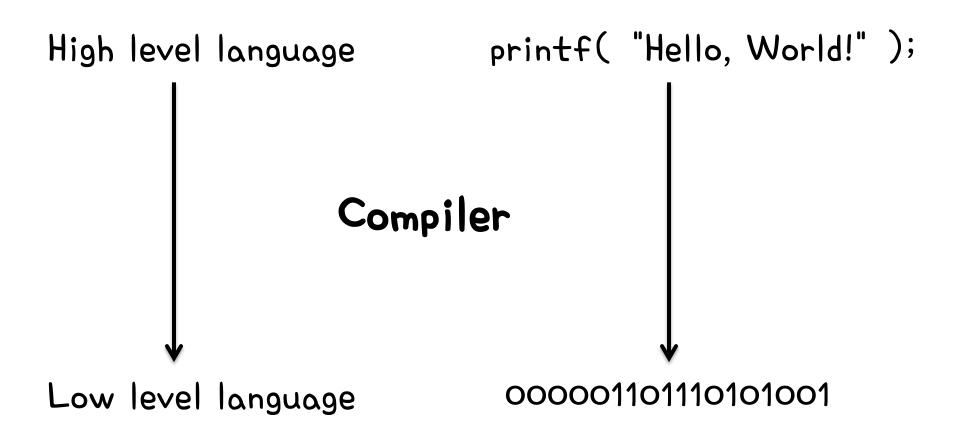
### Programming Language





#### Compiler

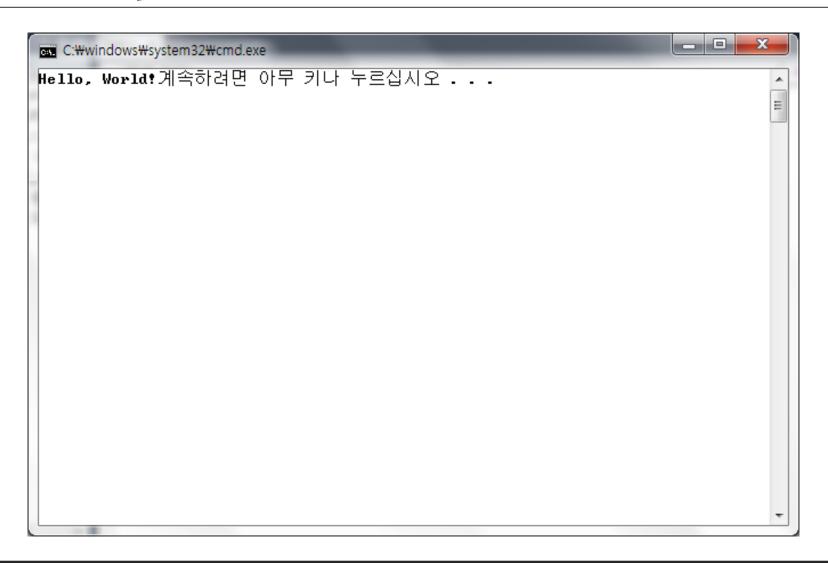






```
#include <stdio.h>
void main()
     printf("Hello, World!");
```

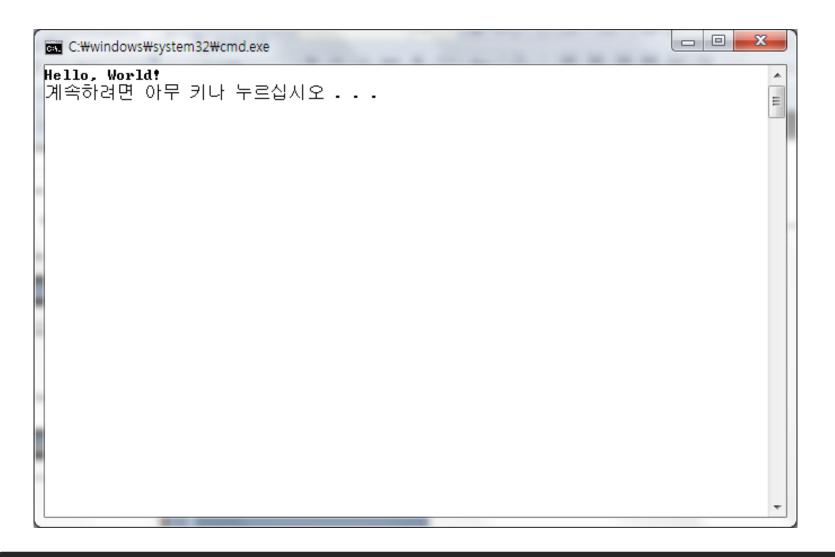






```
#include <stdio.h>
void main()
     printf("Hello, World!\n");
```







```
#include <stdio.h>
void main()
     int a;
     a = 10;
     printf( "a 의 값은: %d ₩n" ,a);
```



```
C:\windows\system32\cmd.exe
a 의 값은 : 10
계속하려면 아무 키나 누르십시오 . . . _
```



```
char
  - 1byte
  - 문자 (%c)
int
  - 4byte
  - 정수 (%d)
float
  - 4byte
  - 실수 (%f)
```



#### 변수 이름 만들기

- 영문자(대문자/소문자), 숫자, '\_' 사용 가능 (대문자와 소문자는 다르게 취급)
- 글자는 영문자나 '\_' 로 시작
- 띄어쓰기 X
- 14자를 넘을 수 없음
- C에 내장되어 있는 단어 사용X (break, case, char, const, continue,…)
- 변수 이름으로 어떤 변수인지 유추 가능하도록 (firstNum, studentName, n\_student, …)

### Printf



```
#include <stdio.h>
void main()
{
    int a;
    a = 13;
    printf( "안녕하세요 %d 학번 여러분 \\mathbb{W}n ", a);
}
```

```
%d, %x, %o : 정수(10진수, 16진수, 8진수)
%f, %lf : 실수
%c : 문자
```

₩n : 새로운 줄로 이동 (Enter 키 )

₩t : 다음 탭으로 이동 ( Tab 키 )

₩b : 뒤로 한칸 이동 ( ← 키 )

₩r : 줄의 맨 앞으로 이동 ( Home 키 )

#### Practice



Printf 함수와 줄바꿈 문자를 이용해서 트리를 만들어 보세요.

```
- 0
C:₩windows₩system32₩cmd.exe
| !!
|계속하려면 아무 키나 누르십시오 . . . _
```



### C programming을 한다?

- -> **함수를** 디자인 한다. (**함수 == 기능**)
- -> void main()

Program을 실행하면 가장 먼저 호출되는 함수가 바로 main입니다.

-> 함수를 디자인 한다. ..??

printf(), scanf() 는 만든 적 없는데.. 어떻게 쓰지?

이미 만들어진 표준 입출력 함수들이랍니다.

#include <stdio.h>를 쓰는 이유는?!!!!

#### 헤더 파일 선언



```
- | ∈ 📦
(전역 범위)
≡#include <stdio.h>
⊡void main(
                                       함수를 쓰기 위해서는
      int ny/m/1 //
     floa/t /fn/m1 = 1.1, fnum2 = 2.2; 그 함수에 대한
               3, num2 = 2;
                                       정보가 필요합니다!!
     scan∱("<mark>//</mark>#", &num1);
                                       그 정보는, header file에
                                       포함 되어있어요
     scanf(<mark>/ %</mark>f", &fnum1);|
     printf∰3 나누기 2는 %d ₩n", num1 / num2);
     print∱("3 나누기 1.1은 %f ₩n", num1 / fnum2);
     printf("1.1 나누기 2.2는 %f ₩n", fnum1 / fnum2);
```

### Scanf



```
#include <stdio.h>
void main()
      int a;
      printf( "숫자를 입력하세요 : ");
      scanf( "%d") ,(&a);
      printf( "입력받은 숫자는 %d₩n" ,a);
                                   대입시킬 변수의 이름
     입력받을 형태 (%d, %c, %f)
```

#### Arithmetic Operations



```
#include <stdio.h>
void main()
        int a, b;
        int sum, sub, mul;
        float div;
        sum = a+b;
        sub = a-b;
        mul = a*b;
        div = (float)a/b;
        printf( "더하기: %d, 빼기: %d₩n" ,sum,sub);
        printf( "곱하기: %d, 나누기: %f₩n ",mul,div);
```

#### Operation



컴퓨터는 <mark>멍청</mark>합니다. 여러 가지를 한번에 못해요.

Num = 
$$5 + 3 + 2$$
;  
Num =  $(5 + 3) + 2$ ;  
Num =  $(8 + 2)$ ;

#### Operation



연산자는 값을 반환 합니다.

```
Num1 = 5 + 3;  //5 + 3은 8이라는 값을 반환
Num1 = 8;  // = 도 연산자임. (대입)
  // = 는 대입한 결과를 반환.
그렇다면
Num2 = Num1 = 5 + 3; 은…?
```

#### Type casting



방금전 (float) 왜 붙이는지 이해 했나요?

(자료)형 변환 에는 자동 형 변환과 강제 형 변환이 있습니다.

자료형? Float/ int / char …

#### Type casting (자동 형 변환 첫 번째 케이스)



```
int num;
float fnum;
```

```
num = 1.1 + 2.2; // 3.3의 결과가 3으로..
fnum = 1 + 2; // 3의 결과가 3.00..으로
```

대입 연산의 대상을 기준으로 변한다! ★

### Type casting example



```
(전역 범위)
≡#include <stdio.h>
⊟void main(){
     int num:
     float fnum;
     num = 1.1 + 2.2;
     fnum = 1 + 2;
     printf("3.3 (float)이 %d 으로..\n\n", num);
     //printf("3.3 (float)이 %f 으로..\n\n", num);
     printf("3(int)가 %f 으로..\n\n", fnum);
     //printf("3(int)가 %d 으로..\n\n", fnum);
```

#### Type casting (자동 형 변환 두 번째 케이스)



```
int num1 = 1, num2 = 2;
float fnum1 = 1.1, fnum2 = 2.2;
num1 + fnum1 //결과는 float형 2.1
num1 + num2 //결과는 int형 3
fnum1 + fnum2 //결과는 float형 3.3
```

손해를 덜 보는 방향으로 자동 형 변환!!

#### example

(전역 범위)

```
⊟#include <stdio.h>
⊟void main(){
     int num1 = 1, num2 = 2;
     float fnum1 = 1.1, fnum2 = 2.2;
     printf("num1 + fnum1 = %f \nskrup", num1 + fnum1);
     //printf("num1 + fnum1 = %d \n", num1 + fnum1);
     printf("num1 + num2 = %d \forall num1 + num2);
     //printf("num1 + num2 = %f ₩n", num1 + num2);
     printf("fnum1 + fnum2 = %f \nskrup", fnum1 + fnum2);
     //printf("fnum1 + fnum2 = %d \mathfrak{"n", fnum1 + fnum2);
     printf("\n");
     num1 = 1.1 + 2.2;
     printf("float의 결과가 int로.. %d \n", num1);
     //printf("float의 결과가 int로.. %f \n", num1);
     fnum1 = 1 + 2;
     printf("int의 결과가 float으로.. %f \n", fnum1);
     //printf("int의 결과가 float으로.. %d \n", fnum1);
```



#### Type casting



```
int num1 = 1, num2 = 2;
float fnum1 = 1.1, fnum2 = 2.2;
num1 / fnum1 //결과는 float
num1 / num2 //결과는??? int!! zero
fnum1 / fnum2 //결과는 float
```

#### Type casting



해결책은?? 강제 형 변환!

num1 / (float) num2 (float) num1 / num2

둘 중 하나만 float이면, (손해를 덜 보는 방향으로!!) 자동으로 나머지 피 연산자 하나까지 float화.



```
🕶 🕬 main()
(전역 범위)
⊞#include <stdio.h>
⊟void main(){
     int num1 = 3, num2 = 2;
     float fnum1 = 1.1, fnum2 = 2.2;
     printf("3 나누기 2는 %d \n", num1 / num2);
     //printf("3 나누기 2는 %f \n", num1 / num2);
     //결과가 int인데 %f로 출력하려 하기 때문에, 0.0000으로 나온다.
     printf("3 나누기 1.1은 %f ₩n", num1 / fnum2);
     printf("1.1 나누기 2.2는 %f \mun", fnum1 / fnum2);
```



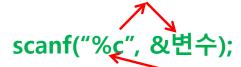
#### **Output**

```
(전역 범위)
≡#include <stdio.h>
     int num1 = 1, num2 = 2;
     float fnum1 = 1.1, fnum2 = 2.2;
     printf("num1 + fnum1 = %f \mathfrak{W}n", num1 + fnum1);
     //printf("num1 + fnum1 = %d \mathfrak{w}n", num1 + fnum1);
     printf("num1 + num2 = %d \forall n", num1 + num2);
     //printf("num1 + num2 = %f \mathfrak{m}n", num1 + num2);
     printf("fnum1 + fnum2 = %f \forall n", fnum1 + fnum2);
     //printf("fnum1 + fnum2 = %d \n", fnum1 + fnum2);
     printf("\n");
     num1 = 1.1 + 2.2;
     printf("float의 결과가 int로.. %d \n", num1);
     //printf("float의 결과가 int로.. %f \n", num1);
     fnum1 = 1 + 2;
     printf("int의 결과가 float으로.. %f \n", fnum1);
     //printf("int의 결과가 float으로.. %d \n", fnum1);
```









```
(전역 범위)
≡#include <stdio.h>
                                                                                                                                                             입력: a ₽
                                                                                   a′₩n′ ◀
⊡void main(){
     int num1 = 1, num2 = 2;
     float fnum1 = 1.1, fnum2 = 2.2;
     printf("num1 + fnum1 = %f \mathfrak{W}n", num1 + fnum1);
     //printf("num1 + fnum1 = %d \mathcal{W}n", num1 + fnum1);
     printf("num1 + num2 = %d \mathfrak{w}n", num1 + num2);
     //printf("num1 + num2 = %f \mathfrak{\pi}n", num1 + num2);
                                                                                 buffer
                                                                                                                                               S
                                                                                                                                                    D
                                                                                                                                                              G
     printf("fnum1 + fnum2 = %f \forall n", fnum1 + fnum2);
     //printf("fnum1 + fnum2 = %d \n", fnum1 + fnum2);
     printf("\n");
     num1 = 1.1 + 2.2
     printf("float의 결과가 int로.. %d \n", num1);
     //printf("float의 결과가 int로.. %f \n", num1);
                                                                                                                                                             입력: 10 ←
     printf("int의 결과가 float으로.. %f \n", fnum1);
     //printf("int의 결과가 float으로.. %d \n", fnum1);
                                                                                  10′₩n′ ←
```

Scanf("%d", &변수);



```
ConsoleApplication6
                                              (전역
      #define _CRT_SECURE_NO_WARNINGS
      #include <stdio.h>
     ⊡void main()
          char d;
          char c;
          printf("알파벳을 입력하시오: ");
          scanf("%c",&d);
          printf("알파벳을 입력하시오: ");
          scanf("%c", &c);
          printf("1번 %c \n2번: %c",d,c);
```



```
`벳을 입력하시오: f
알파벳을 입력하시오: 1번 f
계속하려면 아무 키나 누르십시오 . .
```



```
#define _CRT_SECURE_NO_WARNINGS
 #include <stdio.h>
□void main()
     char d;
     char c;
     char temp;
     printf("알파벳을 입력하시오: ");
     scanf("%c",&d);
     scanf("%c",&temp);//'\n' 을 제거 하기위한 임시 변수
     printf("알파벳을 입력하시오: ");
     scanf("%c", &c);
     printf("1번 %c \n2번: %c",d,c);
```



```
알파벳을 입력하시오: d
알파벳을 입력하시오: f
계속하려면 아무 키나 누르십시오 . . .
```

#### Practice



세 원의 반지름을 입력 받아 두 원의 넓이의 합과 평균을 구해 보세요

```
첫번째 원의 반지름:
5
두번째 원의 반지름:
3
세번째 원의 반지름:
6
세원의 넓이의 합은 87.92 입니다.
세원의 넓이의 평균 29.306665 입니다.
```

## Thank You.

