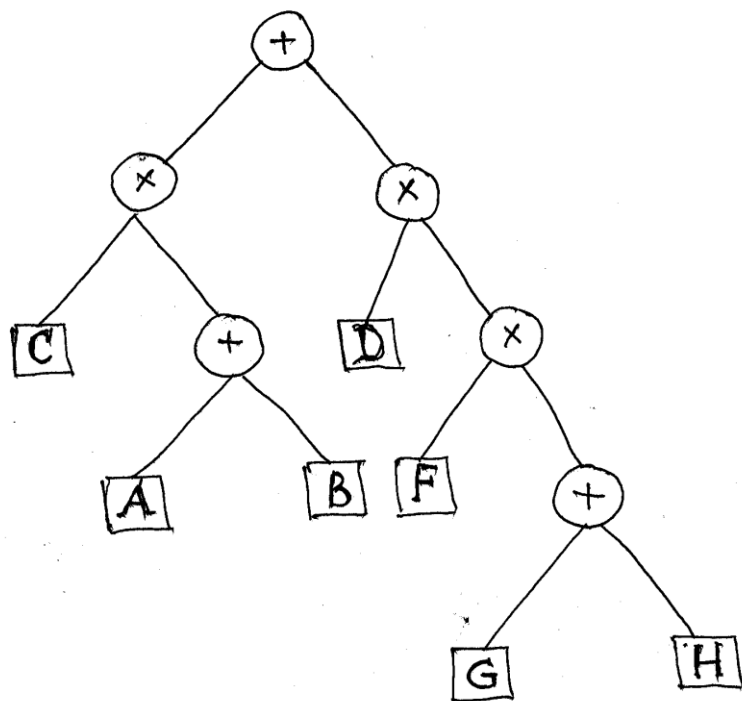




# 시스템프로그래밍기초 실습 5주차

---

# Infix, Prefix, Postfix



- infix:

$(C \times (A + B)) + (D \times (F \times (G + H)))$

- prefix:

$+ \times C + A B \times D \times F + G H$

- postfix:

$C A B + \times D F G H + \times \times +$

# 문자열 입력 함수 - scanf, gets, fgets

- **scanf**: 주어진 문자열 스트림 소스에서 지정된 형식으로 데이터를 읽어내는 기능.  
※ 공백을 입력받지 못함.

```
int scanf(const char *format, ...);
```

- **gets**: 표준 입력에서 줄을 읽고 호출에 의해 버퍼로 불러와 저장하는 기능.  
※ 줄바꿈을 입력받지 못함.

```
char *gets(char *str);
```

- **fgets**: 파일 스트림으로부터 줄을 읽고 저장하는 기능.

```
char *fgets(char *str, int size, FILE *stream);
```

# 실습 예제 1) string\_input.c

- scanf, gets, fgets 비교

```
1 #include <stdio.h>
2
3 int main(void) {
4     // initialize the string
5     char str[100] = {0,};
6     printf("Initial str: %s\n\n", str);
7
8     printf("Now starting the scanf function\n");
9     printf("scanf input:");
10    scanf("%s", str);
11    printf("scanf str: %s\n\n", str);
12
13    printf("Now starting the gets function\n");
14    printf("gets input:");
15    gets(str);
16    printf("gets str: %s\n\n", str);
17
18    printf("Now starting the fgets function\n");
19    printf("fgets input:");
20    fgets(str, sizeof(str), stdin);
21    printf("fgets str: %s\n\n", str);
22
23    return 0;
24 }
```



# 실습 예제 1) string\_input.c 결과

```
inshik@inshik-cpslab:~/Dropbox/yang/lecture/시스템프로그래밍기초2018/실습자료/실습코드/05$ gcc -o string_input string_input.c
string_input.c: In function 'main':
string_input.c:15:2: warning: implicit declaration of function 'gets'; did you mean 'fgets'? [-Wimplicit-function-declaration]
  gets(str);
  ^~~~~
  fgets
/tmp/ccmFQMMU.o: In function 'main':
string_input.c:(.text+0xc3): warning: the 'gets' function is dangerous and should not be used.
inshik@inshik-cpslab:~/Dropbox/yang/lecture/시스템프로그래밍기초2018/실습자료/실습코드/05$ ./string_input
Initial str:

Now starting the scanf function
scanf input:abc def gh
scanf str: abc

Now starting the gets function
gets input:gets str:  def gh

Now starting the fgets function
fgets input:1234 qwer asdf
fgets str: 1234 qwer asdf
```

```
inshik@inshik-cpslab:~/Dropbox/yang/lecture/시스템프로그래밍기초2018/실습자료/실습코드/05$
```

# 문자열 조작 함수(string.h)

**char \*strcat(char \*s1, const char \*s2)**

- 두 문자열 s1, s2를 결합하고, 결과는 s1에 저장

**int strcmp(const char \*s1, const char \*s2)**

- s1과 s2를 사전적 순서로 비교하여, s1이 작으면 음수, 크면 양수, 같으면 0을 리턴

**char \*strcpy(char \*s1, const char \*s2)**

- s2의 문자를 \0이 나올 때 까지 s1에 복사

**size\_t strlen(const char \*s)**

- \0을 뺀 문자의 개수를 리턴

## 실습 예제 2) func\_str.c

```
spubuntu@sp: ~/sysPro/spWeek5
1 #include <stdio.h>
2 #include <string.h> // strcat, strcmp, strcpy, strlen
3
4 int main()
5 {
6     char str1[10];
7     strcpy(str1, "STAR");
8     char *str2 = "BUCKS";
9     printf("str1: %s\t\tstr2: %s\n\n", str1, str2);
10
11     printf("Do strcat(str1, str2)\n");
12     strcat(str1, str2);
13     printf("str1: %s\t\tstr2: %s\n\n", str1, str2);
14     printf("strlen(str1): %lu\t\tstrlen(str2): %lu\n\n",
15           strlen(str1), strlen(str2));
16
17     printf("str1[3] string: %s\t char: %c\n\n", &str1[3], str1[3]);
18     printf("str1[9] string: %s\t char: %c\n\n", &str1[9], str1[9]);
19
20     printf("Do strcmp(str1, str2)\n");
21     if (strcmp(str1, str2) == 0)
22         printf("str1 is equal to str2!\n");
23     else
24         printf("str1 is not equal to str2...\n");
25
26     return 0;
27 }
```

## 실습 예제 2) func\_str.c 결과

```
spubuntu@sp:~/sysPro/spWeek5$ gcc -o func_str func_str.c
spubuntu@sp:~/sysPro/spWeek5$ ./func_str
str1: STAR                str2: BUCKS

Do strcat(str1, str2)
str1: STARBUCKS          str2: BUCKS

strlen(str1): 9           strlen(str2): 5

str1[3] string: RBUCKS    char: R
str1[9] string:           char:

Do strcmp(str1, str2)
str1 is not equal to str2...
```



# 실습 과제 1) my\_str.c

- 문자열을 사용해 strcmp와 strcat을 직접 만들어볼 것.

```
spubuntu@sp: ~/sysPro/spWeek5
1 #include <stdio.h>
2
3 char *my_strcat(char *s1, const char *s2)
4 {
5     /* to be implemented */
6
7     return s1;
8 }
9
10 int my_strcmp(const char *s1, const char *s2)
11 {
12     int c = 0;
13     /* to be implemented */
14 }
15
16 int main()
17 {
18     char *str1 = "STARBUCKS";
19     char str2[10] = "STAR";
20     char *str3 = "BUCKS";
21
22     printf("str1: %s\t\tstr2: %s\t\tstr3: %s\n", str1, str2, str3);
23     printf("my_strcmp(str1, str2): %d\n", my_strcmp(str1, str2));
24     printf("my_strcmp(str3, str2): %d\n\n", my_strcmp(str3, str2));
25
26     my_strcat(str2, str3);
27
28     printf("str1: %s\t\tstr2: %s\t\tstr3: %s\n", str1, str2, str3);
29     printf("my_strcmp(str1, str2): %d\n", my_strcmp(str1, str2));
30
31     return 0;
32 }
```

# 실습 과제 1) my\_str.c 결과

```
spubuntu@sp:~/sysPro/spWeek5$ ./my_str
str1: STARBUCKS      str2: STAR      str3: BUCKS
my_strcmp(str1, str2): 1
my_strcmp(str3, str2): -1

str1: STARBUCKS      str2: STARBUCKS      str3: BUCKS
my_strcmp(str1, str2): 0
spubuntu@sp:~/sysPro/spWeek5$
```

## 실습 과제 2) script\_week5

- 기존의 script와 같이 예제 및 과제 파일을 컴파일 및 실행하여라.
- infix로 표현된 아래 식을 script 내에 prefix, postfix로 출력하여라.

$$A + B \% C \times ( D + E / ( F - G ) * H )$$

※ script 파일 마지막에 다음 명령을 추가해 출력할 것.

```
echo "infix: [ A + B % C x ( D + E / ( F - G ) * H ) ]"  
echo "prefix: [예상값]"  
echo "postfix: [예상값]"
```

# 과제 제출 파일

1. 모든 파일은 sys\_05 학번.tar.gz으로 압축하여 제출한다.

1. string\_input.c

2. str\_func.c

3. my\_str.c

4. script\_week5



# 감사합니다.

---