

프론트 13th 수업에서 다루어지는 것들 문자열(스트링, string)

문자 vs 문자열

문자(character): 각각의 문자에 숫자코드(아스키코드)가 할당되어 있다. 예: 'A'(65), 'a'(97), '0'(
문자열(string): 문자들이 여러 개 모인 것

문자열 상수 vs 문자열 변수

문자열 상수: "A", "Hello World"

문자열 변수: char str[100]; 로 선언하고 값을 넣을 때 마지막에 '\0'이 들어가야 문자열 변수

문자열 vs 문자형 배열

문자열은 끝에는 항상 NULL문자('\0')가 들어가는 문자형 배열
NULL은 문자열의 끝을 알려준다.

문자열 변수의 선언 및 초기화

```
char str[10];  
char str[10] = {'a', 'B', 'c', 'D', 'e', '\0'};  
char str[10] = "aBcDe"; // 자동으로 '\0'을 붙여준다
```

```
char str[] = "aBcDe"; // str[6]로 자동 설정된다
```

문자열 변수의 입력

```
scanf("%s", str); // 문자열 변수의 입력에서는 &를 붙이지 않는다. 사실은 str == &str[0]
```

문자열 변수의 출력

방법 1:

방법 2:

문자열 변수 안의 문자들을 처리

```
char str[] = "aBcDe";
```

```
for(i = 0; str[i] != '\0'; i++)  
    한 문자씩 처리;
```

□ 연습 1: str의 홀수번째 문자를 출력

실행예
ace

```
for(i = 0; str[i] != '\0'; i++)
```

□ 연습2: str안에 대문자를 출력

실행예
BD

```
for(i = 0; str[i] != '\0'; i++)
```

□ 연습2-2: str안에 대문자를 가지고 str2를 만들어 str2를 출력

실행예
BD

```
for(i = 0; str[i] != '\0'; i++)  
{
```

```
}  
printf("%s", str2);
```

□ 연습3: str안에 대문자를 소문자로 소문자를 대문자로 출력

실행예
AbCdE

```
for(i = 0; str[i] != '\0'; i++)
```

□ 연습3-2: str안에 대문자를 소문자로 소문자를 대문자로 바꾸어 str2를 만들어 str2를 출력

실행예
AbCdE

```
for(i = 0; str[i] != '\0'; i++)
```

□ 연습4: 문자열 str의 길이를 출력

실행예
5

```
for(i = 0; str[i] != '\0'; i++)
```

□ 연습5: str을 거꾸로 출력

실행예
eDcBa

□ 연습5-2: str을 거꾸로 한 문자열을 str2에 넣어서 str2를 출력

실행예
eDcBa

문자열의 함수 매개변수 전달

```
#include <stdio.h>  
void printUpperCase(char s[]);  
int main(void)  
{  
    Char str[10] = "aBcDeF";  
    printUpperCase(    ); // 문자열 변수의 이름  
}
```

```
void printUpperCase(char s[]) // 배열의 경우는 크기를 같이 전달했으나, 문자열은 그럴 필요 없음. 왜?  
{
```

LAB 13B — 문자열

☞오늘의 실습숙제에서
(수업시간에 배우지 않은) 문자열 처리함수(strlen, strcmp 등)나
문자처리함수(isupper, islower등)는 사용하지 않는다.

■ **LAB13B_0**(문자열의 정의, 표준출력, 문자열 종료방법) 다음 예제 프로그램의 결과 예상해 보세요.
그리고 프로그램의 실행한 후 자신의 예상 결과와 비교해 보세요. 1)2)3)4)를 잘 살펴보라.

```
#include <stdio.h>
int main(void)
{
    int i;

    // 1) 문자열을 정의하는 여러가지 방법들입니다
    char digits[] = "0123456789";
    char abc[] = {'A', 'B', 'C', '\0'};
    char lan[5] = "java";

    // 2) 출력 결과는?
    printf("%s %s\n", digits, abc); //%s를 사용한다.

    // 3) 아래처럼 lan 문자열변수를 두가지 방법으로 출력할 수있다. %c와 %s의 쓰임새에 주목하라.
    printf("%s\n", lan);

    for (i = 0; lan[i] != '\0'; i++) // A)문자열의 마지막을 확인하는 방법을 유의해보라
        printf("%c", lan[i]);

    // 4) digits[]의 중간에 null character를 삽입하면
    digits[6] = '\0';
    printf("\n%s\n", digits);

    return 0;
}
```

■ LAB13B_a

- 하나의 문자열 변수 word를 읽어들어서(%s사용) 이를 출력(%s사용)하는 간단한 프로그램을 작성하라. word의 길이는 최대 10이라 가정하자. 즉 char word[11]; 이라 선언하여 사용하면 된다.

실행예:
Enter a word(<= 10 chars): Happy
Happy

- 위의 프로그램을 수정하여 출력할 때 word안의 문자를 하나하나를 출력하도록 하라. 즉, 실행예는 그대로이다. (위의 LAB13B_0의 A)부분을 참조하라)

- 다시 위의 프로그램을 수정하여 word의 요소중 홀수번째(즉, 첫번째, 세번째,...)의 문자를 출력하도록 하라.

실행예
Enter a word(<= 10 chars): Happy
Hpy

실행예
Enter a word(<= 10 chars): Love
Lv

- **LAB13B_1**(문자열 처리) 문자열(최대 길이 80라 가정하자)을 입력받아서, 입력받은 문자열의 길이를 출력하고 그 문자열을 뒤에서부터 한줄에 한 단어씩 출력하는 프로그램을 작성하라.

- **LAB13B_1_1**(길이 구하기) 먼저 문자열의 길이를 구하여 출력하는 프로그램을 작성하자.

실행예:

```
Enter a string: abcde
길이는 5
```

- s라는 이름의 문자열 변수를 정의하자. 크기는 얼마로 지정해야하는가.

- 문자열 변수의 문자들을 하나씩처리하기 위한 for문은 아래와 같다.
for(i = 0; s[i] != '\0'; i++)

아래와 같이 loop body부분 없는 for문을 수행하면 for문 이후 i는 어떤 값을 갖는 가를 생각해보라.
for(i = 0; s[i] != '\0'; i++);

- 자, 프로그램을 작성해보자.

```
#include <stdio.h>
int main(void)
{
```

```
    //필요한 변수
```

```
    // 문자열 변수 입력
```

```
    // 문자열 변수의 길이를 센다(for문 사용)
```

```
    // 길이를 출력한다.
```

```
}
```

- (길이를 이용하여 거꾸로 쓰기) 위에서 구한 문자열의 길이를 이용하여 아래의 실행결과를 내도록 위의 프로그램을 수정하라.

실행예:

```
Enter a string: abcde
길이는 5
e
d
c
b
a
```

힌트: 길이(length)를 구하면 문자열의 마지막 요소부터의 출력이 가능하다.

예: abcde의 경우 길이는 5이고 s[4], s[3], s[2], s[1], s[0]으로 출력하면 된다.

□ **LAB13B_1_2**(palindrome 만들기 준비)

```
#include <stdio.h>
int main(void)
{
    int i;
    char ch;
    char str[10] = "abcde"; // 1)

    printf("--변경 전 문자열--\n");
    printf("%s \n", str);

    /* 문자열 변경 */
    for(i=0; i < 2; i++) // 2)
    {
        ch = str[4 - i]; // 3)
        str[4 - i] = str[i]; // 4)
        str[i] = ch;
    }

    printf("\n--변경 후 문자열--\n");
    printf("%s \n", str);

    return 0;
}
```

- ☆ 프로그램을 이해할 수 있는가? 메모리에 값저장 상태를 나타낸 후 2)의 for문에서 반복이 이루어 질때마다 메모리 상의 어떤 변화가 일어나는지 그려보라.

초기 상태

| | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|
| | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|

첫번째 반복: i가 0일 때

| | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|
| | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|

두번째 반복: i가 1일 때

| | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|
| | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|

- ☆ 1)에 있는 초기화부분을 다음과 같이

```
char str[10] = "12345678";
```

로 바꾼 후 아래의 실행결과가 나오도록 프로그램을 수정하라.

힌트: 라인 2), 3), 4)에서 숫자 상수만 바꾸면 된다.

```
C:\ "C:\WDocuments and Settings\W
--변경 전 문자열--
12345678
--변경 후 문자열--
87654321
Press any key to continue
```

■ **LAB13B_2**(문자열처리)

- **(LAB13B_2_1)**문자열을 입력받아서, 그 안에 존재하는 대문자 알파벳들을 출력하는 프로그램을 작성하라.

| | |
|----------------------------------|-------------------------------------|
| C:\ "F:\W++\201001수업W-2010컴프1강의자 | C:\ "F:\W++\201001수업W-2010컴프1강의자 |
| Enter one word: DongDuk2010A | Enter one word: VeryImportantPerson |
| D | V |
| D | I |
| A | P |
| Press any key to continue_ | Press any key to continue_ |

```
#include <stdio.h>
int main(void)
{
    //필요한 변수

    // 문자열 변수 입력

    // 문자열 변수안 문자를 하나씩 처리하면서 대문자 알파벳이면 출력한다. 대문자 알파벳인가를
    // 판별하기위해서 ASCII 값을 이용한다. 문자 'A'와 문자 'Z'의 ASCII값은 각각 65, 90이다.
    // 즉 65이상 90이하이면 대문자 알파벳 임을 알수있다.

}
```

- **(LAB13B_2_2)**위의 프로그램을 수정하여, 문자열 word를 입력받아서 word 안의 대문자들을 다른 문자열 변수 newWord에 넣은 후 이것을 출력하는 프로그램을 작성하라.

| | |
|--------------------------------|-----------------------------------|
| C:\ "F:\W++\201001수업W-2010컴프1강 | C:\ "F:\W++\201001수업W-2010컴프1강의자료 |
| Enter one word: DongDuk82AbC | Enter one word: AsSoonAsPossible |
| DDAC | ASAP |
| Press any key to continue | Press any key to continue_ |

힌트 및 요구사항

- newWord 문자열 변수를 위한 별도의 첨자, j(0으로 초기화)가 필요하다.
- 출력시 아래와 같은 이상한 문자들이 올바른 실행결과 뒤에 따라나올 수있다. 그런 경우 왜일 까를 생각해보라. 어떻게 해결할것인가?

<잘못된 실행예>

| |
|--|
| C:\ "F:\W++\201001수업W-2010컴프1강의자료W실습숙제답W실습숙제10답WLAB11_2_2... |
| Enter one word: DongDuk82A |
| DDA |
| ongDuk82A |
| Press any key to continue_ |

```
#include <stdio.h>
int main(void)
{
    char word[81], newWord[81];

    printf("%s\n", newWord);
}
```

HW 13B

- **HW13B_1**(문자열 처리) 하나의 문자열(80자이하)을 입력 받아서 문자열 안의 대문자는 소문자로, 소문자는 대문자로 바꾸는 프로그램을 작성하세요. 영문자 이외의 숫자나 기호는 변환하지 않습니다.

힌트 및 요구사항

- 새로 변환한 문자열을 반드시 새로운 문자열 에 저장하라.
즉, 입력 받을 문자열은 word, 새로 변환한 문자열은 newWord를 사용한다. 두 문자열 모두 char형이고 크기는 81로 한다.
- 대문자 알파벳과 소문자 알파벳은 각각 32의 차이값을 갖는다. 즉, 'A' 는 65이고 'a' 는 97 값을 갖는다. 그러므로
= 대문자를 소문자로 바꾸는 방법: 대문자 + 32
= 소문자를 대문자로 바꾸는 방법: 소문자 - 32
- 새로 변환한 문자열 출력시
printf("%s" , newWord)를 사용하라.

```
GNU "F:\W++\201001수업W-)2010컴프1
```

```
Enter one word: WordCup2010
word given: WordCup2010
new word: wORDcUP2010
Press any key to continue
```

```
GNU "F:\W++\201001수업W-)2010컴프1
```

```
Enter one word: DongDuk2010
word given: DongDuk2010
new word: dONGdUK2010
Press any key to continue
```

- **HW13B_2**(문자열) 적당한 길이의 문자열을 입력받아서, 그 안에 존재하는 숫자들의 총합을 계산하여 출력하는 프로그램을 작성하라. '0' 의 아스키값은 48임을 활용하라.
예를 들어, '1' 은 49값을 갖으므로 '1' - 48 계산을 통해서 1이라는 숫자 값을 얻을 수있음에 착안하라.

```
GNU "F:\W++\201001수업W-)2010컴프1
```

```
Enter one word: ab3cd4e56f
안에 있는 숫자들의 합은 18
Press any key to continue
```

```
GNU "F:\W++\201001수업W-)2010컴프1
```

```
Enter one word: Dongduk2010
안에 있는 숫자들의 합은 3
Press any key to continue
```

- **HW13B_3**(문자열 palindrome만들기) 문자열(word)을 입력받아 그 속의 문자들을 거꾸로 하여 문자열(newWord)을 만들어 출력하는 프로그램을 작성하라.(LAB13B_1을 참조)

실행예 :

Enter one word: abcde

The reversed word is edcba

```
#include <stdio.h>
int main(void)
{
    char word[81], newWord[81];
```

```
    printf("%The reversed word is s\n", newWord);
}
```

문자열은 다양한 문제에서 연습이 가능합니다.
실제 프로그래밍에서 문자열을 다루는 연습은 아주 중요합니다.

아래의 challenge 문제들은 수업뒤에서 연습문제로 사용될 수도 있습니다.

- **challenge13a**(난이도 상) 두 개의 단어를 입력받아서 같은가 다른가를 판별하는 프로그램을 작성하라. 아래의 실행결과를 모두 실행시켜보아 프로그램의 완성도를 확인하라.

```
C:\ "F:\W++\201001수업W-)2010컴>
Enter the first word: Park
Enter the second word: Pak
두 단어는 다르다
Press any key to continue_
```

```
C:\ "F:\W++\201001수업W-)2010컴>
Enter the first word: Pak
Enter the second word: Park
두 단어는 다르다
Press any key to continue
```

```
C:\ "F:\W++\201001수업W-)2010컴>
Enter the first word: Pak
Enter the second word: Pakk
두 단어는 다르다
Press any key to continue
```

```
C:\ "F:\W++\201001수업W-)2010컴>
Enter the first word: Pakk
Enter the second word: Pak
두 단어는 다르다
Press any key to continue_
```

```
C:\ "F:\W++\201001수업W-)2010컴>
Enter the first word: Pak
Enter the second word: Pak
두 단어는 같다
Press any key to continue
```

- **challenge13b**(난이도 상) 하나의 단어를 입력 받아서 이 단어 안에 포함된 숫자의 합을 출력하는 프로그램을 작성하라. 단어 안에 숫자가 연이어 나올 경우 하나의 수로 간주한다.

실행예:
Enter a word: ab22c3d5
글자 안의 수의 합은 30

- **challenge13c**(난이도 중) 주어진 문자열에서 모음-a(A), e(E), i(I), o(O), u(U)-의 개수를 세는 프로그램을 작성하라. 소문자 대문자를 함께 카운트한다.

```
C:\ C:\WINDOWS\system32\cmd.exe
문자열 입력<문자수 81 이하>: abcdefABCDEF
a or A: 2 개
e or E: 2 개
i or I: 0 개
o or O: 0 개
u or U: 0 개
계속하려면 아무 키나 누르십시오 . . .
```

아래의 주어진 프로그램에서 define의 사용을 잘 살펴보라.

```
#include <stdio.h>
```

```
#define MAX_STRING 81
```

```
int main()
{
    char str[MAX_STRING];
    int countA = 0, countE = 0, countI = 0, countO = 0, countU = 0;
    int i;

    printf("문자열 입력 (문자수 %d 이하): ", MAX_STRING);
    scanf("%s", str);
```

```
    return 0;
}
```

■ **challenge13d (난이도 중)** 주어진 단어가 palindrome인지를 판별하는 프로그램을 작성하라.
palindrome은 madam나 abccba처럼 앞에서부터 읽으나 뒤에서부터 읽으나 동일한 단어를 의미한다.

아래의 주어진 프로그램에서 define의 사용을 잘 살펴보라.

| C:\WINDOWS\system32\cmd.exe | C:\WINDOWS\system32\cmd.exe |
|-----------------------------|-----------------------------|
| * Palindrome 체크 | * Palindrome 체크 |
| 문자열 입력<문자수 81 이하>: madam | 문자열 입력<문자수 81 이하>: father |
| "madam" is a Palindrome | "father" isn't a Palindrome |
| 계속하려면 아무 키나 누르십시오 . . . | 계속하려면 아무 키나 누르십시오 . . . |

```
#define MAX_STRING 81
#define BOOL int
#define TRUE 1
#define FALSE 0

// Palindrome 검사
BOOL isPalindrome(char str[]);

int main(void)
{
    char str[MAX_STRING];

    printf("* Palindrome 체크\n\n");
    printf("문자열 입력(문자수 %d 이하): ", MAX_STRING);
    scanf("%s", str);

    if (isPalindrome(str)) // Palindrome 검사
        printf("%s\n is a Palindrome\n\n", str); // "를 출력하려면 \"를 써야
    else
        printf("%s\n isn't a Palindrome\n\n", str);

    return 0;
}

BOOL isPalindrome(char s[])
{
    //문자열 s를(사실은 s는 포인터이지만, 이는 2학기때 배운다)
    //체크해서 palindrome이면 TRUE값을 palindrome아니면 FALSE값을 return한다.

}
```