

Test Case Report – C Programming and Lab HW 2

Find Ewha IP

Microsoft Visual Studio 디버그 콘솔

```
IP주소값 별명 5쌍을 입력하세요.  
10.0.0.1 Sangsoo  
203.255.176.124 eslab  
10.2.0 Soosang  
잘못된 입력입니다.  
10.2.0.1 Soosang  
203.255.176.125 esl  
192.168.1.1 cprog  
이화여자대학교에 속한 주소:  
eslab  
esl  
  
C:\Users\redem\source\repos\2020-2) C Programming a  
이 창을 닫으려면 아무 키나 누르세요...
```

Microsoft Visual Studio 디버그 콘솔

```
IP주소값 별명 5쌍을 입력하세요.  
203.255.12.12 testcase  
11. ccc  
잘못된 입력입니다.  
11.11.11.11 test  
203.255.1.1 case  
203.255.1.2 tttt  
12.12.12.12 cccc  
이화여자대학교에 속한 주소:  
testcase  
case  
tttt  
  
C:\Users\redem\source\repos\2020-2) C Program  
) 종료되었습니다(코드: 0개).  
이 창을 닫으려면 아무 키나 누르세요...
```

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h> // stdio.h 헤더 파일, stdlib.h 헤더 파일, string.h 헤더 파일을 포함한다.
#define MAX_CHAR_SIZE 10000 // MAX_CHAR_SIZE를 10000으로 정의한다.

typedef char* element; // char*를 element라고 정의한다.
typedef struct {
    int n[4];
    element name;
} ipaddress; // 정수 배열과 element name을 가지는 ipaddress라는 이름의 구조체를 정의한다.

element k[5]; // 5개의 인덱스를 갖는 element k를 정의한다.
int m = 0; // 정수 m을 0으로 초기화한다.
int num; // 정수 num을 정의한다.

void newIP(ipaddress *x) {
    for (int l = 0; l < 4; l++) {
        x->n[l] = NULL;
    }
    x->name = NULL;
} // 리턴값이 없는 newIP라는 함수를 정의한다. 이 함수는 ipaddress x의 정수 배열의 값과 element name의 값을 모두
NULL로 만든다.


void error(ipaddress* x) {
    if (num != 4) {
        printf("잘못된 입력입니다.\n");
        input(&x);
    }
    else;
} // 리턴값이 없는 error라는 함수를 정의한다. 이 함수는 num이 4가 아닐 때 잘못된 입력이라는 문구를 출력하고
input함수를 실행한다. 만약 그게 아니라면 함수가 종료된다.

int input(ipaddress* x) {
    x->name = (element)malloc(sizeof(char) * MAX_CHAR_SIZE);
    num = scanf("%d.%d.%d.%d", &x->n[0], &x->n[1], &x->n[2], &x->n[3]);
    scanf("%s", x->name);
    if ((x->n[0] == 203) && (x->n[1] == 255)) {
        k[m++] = x->name;
    }
    error(&x);
} // int값을 리턴하는 함수 input을 정의한다. 구조체 x의 element name을 동적할당하고 x의 정수 배열의 값과 element name의
값을 입력받는다. 만약 주소가 203.255로 시작한다면, 배열 k의 m번째 인덱스는 name을 가르킨다.

int main(void) {
    ipaddress ip;
    printf("IP주소값 별명 5쌍을 입력하세요.\n");
    for (int i = 0; i < 5; i++) {
        newIP(&ip);
        input(&ip);
    }
    printf("이화여자대학교에 속한 주소: \n");
    for (int j = 0; j < 5; j++) {
        if (k[j] != NULL) {
            printf("%s\n", k[j]);
        }
    }
    free(ip.name);
    return 0;
} // int값을 리턴하는 함수 main을 정의한다. Ipaddress ip를 선언하고, newIP 함수로 함수를 초기화하며 input 함수로 ip값을 받
는다. 이 실행은 5번 반복되며 잘못된 입력을 받았을 때마다 한 번 더 실행되어 올바른 ip 주소값 5개를 받는다. 모든 입력
이 끝난 후 이화여자대학교에 속한 ip주소만을 출력한다.

```

Find Capital


 선택 Microsoft Visual Studio 디버그 콘솔

문자열을 입력하세요.

JoJo

JoJo에서 대문자는 JJ입니다.

C:\Users\redem\source\repos\2020-2\C Program
이 창을 닫으려면 아무 키나 누르세요...


 Microsoft Visual Studio 디버그 콘솔

문자열을 입력하세요.

hello

hello에서 대문자는 없습니다.

C:\Users\redem\source\repos\2020-2\C Program
이 창을 닫으려면 아무 키나 누르세요...

 Microsoft Visual Studio 디버그 콘솔

문자열을 입력하세요.

CProgrammingAndLab

CProgrammingAndLab에서 대문자는 CPAL입니다.

C:\Users\redem\source\repos\2020-2\C Program
이 창을 닫으려면 아무 키나 누르세요...

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h> // 헤더 파일 stdio.h, stdlib.h, string.h를 포함한다.
#define MAX_SIZE_CHAR 10000 // MAX_SIZE_CHAR을 10000으로 정의한다.
```

```
typedef char* element;
int i = 0;
int j = 0;
// char*을 element로 정의한다. 정수 i와 j를 선언하고 0으로 초기화한다.
```

```
int find_caps(element caps, const element str);
int find_caps(element caps, const element str) {
    char temp = NULL;
    if (i == strlen(str)) {
        return;
    }
    else {
        if (str[i] >= 65 && str[i] <= 90) {
            temp = str[i++];
            caps[j++] = temp;
            find_caps(caps, str);
        }
        else {
            i++;
            find_caps(caps, str);
        }
    }
}
```

} // 정수를 반환하는 find_caps라는 함수를 정의한다. 초기값이 NULL인 char temp를 정의하여 str[i]가 대문자일 때 str[i]의 값을 caps[j]로 복사하도록 한다. 함수를 한 번 반복할 때마다 i는 1씩 증가하며, i의 값이 str의 문자 개수와 같아질 때 이 함수가 종료된다. 그렇지 않을 때는, find_caps 함수 자신을 불러들이며 재귀적 반복을 수행한다.

```
int main(void) {
    element str = (element)malloc(sizeof(char) * MAX_SIZE_CHAR);
    element caps = (element)malloc(sizeof(char) * MAX_SIZE_CHAR);
    memset(caps, NULL, MAX_SIZE_CHAR);
    printf("문자열을 입력하세요.\n");
    scanf("%s", str);
    find_caps(caps, str);
    if (caps[0] == NULL) printf("%s에서 대문자는 없습니다.\n", str);
    else printf("%s에서 대문자는 %s입니다.\n", str, caps);
    free(str);
    free(caps);
    return 0;
}
```

} // 정수를 반환하는 main함수를 선언한다. element str과 element caps를 동적할당한다. memset함수를 이용하여 caps의 모든 인덱스값을 NULL으로 초기화하였다. 문자열 str을 입력받은 후 find_caps를 이용하여 입력받은 문자열의 대문자를 caps에 저장한다. 만약, str에 대문자가 없어서 caps[0]이 NULL이라면 대문자가 없다는 문구를 출력하고, 대문자가 있다면 str과 caps를 출력한다. 마지막에는 동적할당된 str과 caps를 free로 풀어 주고, 0을 리턴하여 함수를 종료한다.