

# 운영체제 및 실습

분	반	00
학	과	컴퓨터공학과
학	번	201502049
이	름	노 효 근

# 1. 문제 해결 방법

## 1) 소스코드

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

int main(void)
{
    int    i = 0, num = 0;
    double j = 0;
    char   *str = NULL;

    printf("Enter the number: ");
    scanf("%d", &num);

    for(i = 0; i < num; i++){
        j = i/3 + i;
        printf("j = %lf \n", j);
    }

    strcpy(str, "Success!");
    printf("%s\n", str);

    return 0;
}
```

해당 소스코드를 보면 \*str이 NULL로 되어있는 것을 확인 할 수 있다. 따라서 해당 파일을 GDB 디버거로 실행을 해서 display 명령어를 통해 \*str의 값을 확인해보면 다음과 같이 메모리 주소 접근이 불가능하다고 한다.

```
(gdb) break 14
Breakpoint 1 at 0x4006a1: file homework.c, line 14.
(gdb) r
Starting program: /home/u201502049/Desktop/OS/homework
Enter the number: 3

Breakpoint 1, main () at homework.c:15
15      for(i = 0; i < num; i++){
(gdb) display *str
1: *str = <error: Cannot access memory at address 0x0>
```

위와 같은 오류를 바로 잡고자 아래와 같이 코드를 수정하였다.

## 2) 해결방법

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

int main(void)
{
    int    i = 0, num = 0;
    double j = 0;
    char   *str = (char*)malloc(sizeof(char)*8);

    printf("Enter the number: ");
    scanf("%d", &num);

    for(i = 0; i < num; i++){
        j = i/3 + i;
        printf("j = %lf \n", j);
    }

    strcpy(str, "Success!");
    printf("%s\n", str);

    return 0;
}
```

NULL로 설정되어있던 \*str값을 malloc 함수를 통해 메모리를 할당한다.  
이후 gcc로 컴파일하여 GDB 디버거로 실행하여 display 명령어를 통해 \*str과 str 값 모두 확인해 보았다.

```
(gdb) n
21          strcpy(str, "Success!");
1: *str = 0 '\000'
2: str = 0x602010 ""
(gdb) n
22          printf("%s\n", str);
1: *str = 83 'S'
2: str = 0x602010 "Success!"
```

다음과 같이 값이 메모리 할당이 되어 들어가는 것을 확인 할 수 있다.

## 2. 실행 결과 화면

```
u201502049@u201502049:~/Desktop/OS$ ./homework
Enter the number: 3
j = 0.000000
j = 1.000000
j = 2.000000
Success!
```