

운영체제 및 실습

- 1차 과제 -

학번: 201404377

이름: 진승언


```
u201404377@u201404377: ~
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from homework...done.
(gdb) r
Starting program: /home/u201404377/homework
Enter the number: 3
j = 0.000000
j = 1.000000
j = 2.000000

Program received signal SIGSEGV, Segmentation fault.
0x0000000000400705 in main () at homework.c:17
17      strcpy(str, "Success!");
(gdb)
```

=>(3) r명을 내려 로드 된 프로그램을(homework) 실행시킨다. 그럼 다음과 같이 Segmentation fault 오류가 코드 17번째 줄에서 발생함을 알 수 있다. homework의 코드를 봐보면 포인터 str이 Null을 가리키고 있으므로 strcpy(str, "Success!") 함수를 사용할 경우 할당된 주소가 없으므로 에러가 남을 알 수 있다.

```
u201404377@u201404377: ~
(gdb) break 17
Breakpoint 2 at 0x4006f7: file homework.c, line 17.
(gdb) r
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/u201404377/homework
Enter the number: 3
j = 0.000000
j = 1.000000
j = 2.000000

Breakpoint 2, main () at homework.c:17
17      strcpy(str, "Success!");
(gdb) print str= (char *)malloc(sizeof(char))
$1 = 0x602830 ""
(gdb) n
18      printf("%s\n", str);
(gdb) n
Success!
19      return 0;
(gdb) n
20  }
(gdb) n
```

=>(4) 에러난 곳을 수정해서 실행시켜보기 위해 breakpoint를 17로 설정해주고 r로 다시 프로그램을 실행시킨다. 그러면 오

류가 낫던 코드줄이 었던 17에서 break가 걸리고 포인터 str을 print명령어를 통해 바꿔준다.

print str=(char *)malloc(sizeof(char)*100) 이렇게 바꿔준다
그다음 n명령어를 통해 마지막 코드까지 에러 없이 잘 디버깅됨을 볼 수 있다.

```
u201404377@u201404377: ~
1 #include <stdio.h>
2 #include <string.h>
3 #include <stdlib.h>
4
5 int main(void)
6 {
7     int    i = 0, num = 0;
8     double j = 0;
9     char   *str = (char *)malloc(sizeof(char)*100);
10    printf("Enter the number: ");
11    scanf("%d", &num);
12    for(i = 0; i < num; i++)
13    {
14        j = i/3 + i;
15        printf("j = %lf \n",j);
16    }
17    strcpy(str, "Success!");
18    printf("%s\n", str);
19    return 0;
20 }
```

-- 끼워넣기 -- 9,47-57 모두

=>(5) q명령어로 gdb에서 나오고 vi homework.c 명령어로 다시 위 사진과 같이 str을 수정해준다.

```
u201404377@u201404377: ~
u201404377@u201404377:~$ vi homework.c
u201404377@u201404377:~$ gcc -o homework homework.c
u201404377@u201404377:~$ ./homework
Enter the number: 3
j = 0.000000
j = 1.000000
j = 2.000000
Success!
u201404377@u201404377:~$
```

=>(6) gcc -o명령어로 다시 컴파일 후 ./homework로 실행시켜주면 다음과 같이 올바르게 실행됨을 볼 수 있다.