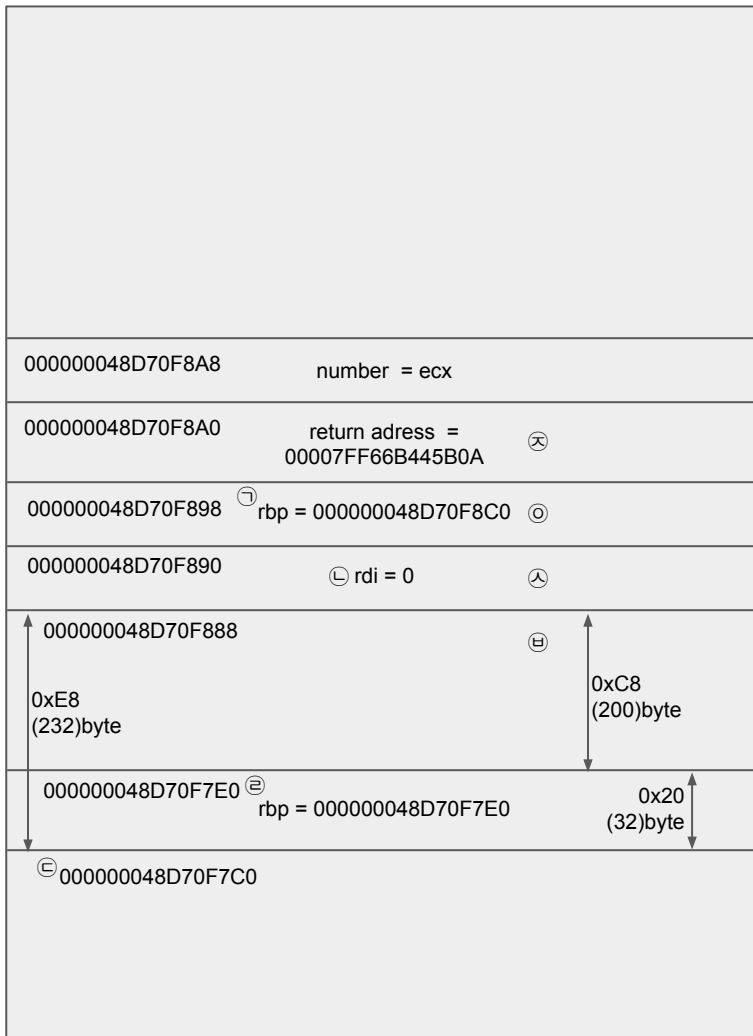


# main

```

㉠ 00007FF66B445AE0 push    rbp
㉡ 00007FF66B445AE2 push    rdi
㉢ 00007FF66B445AE3 sub     rsp,128h
㉣ 00007FF66B445AEA lea     rbp,[rsp+20h]
    ㉤ int return_value;
    ㉥ const int input_parameter = 211;
㉦ 00007FF66B445AFB mov     dword ptr [input_parameter],0D3h
㉧ 00007FF66B445B02 mov     ecx,dword ptr [input_parameter]
㉨ 00007FF66B445B05 call    for_assembly_function_test (07FF66B4411CCh)
㉩ 00007FF66B445B0A mov     dword ptr [return_value],eax
  
```

- ㉠ rbp값을 초기화 하고 저장한다.
- ㉡ rdi값을 초기화하고 저장한다.
- ㉢ rsp로부터 0x128(296)byte만큼 Stack공간을 확보한다.  
이 때, 이 공간은 변수를 저장하기 위한 공간으로 활용된다.
- ㉣ rsp로부터 0x20(32)byte만큼 위쪽에 rbp값을 새로 지정한다.
- ㉤ ㉤에서 확보한 공간 맨 아래에 return\_value라는 int 변수를 저장한다.
- ㉥ ㉤에서 확보한 공간 중 가장 마지막으로 선언된 변수에서 32byte만큼 위쪽에 input\_parameter라는 const int변수를 저장한다.
- ㉦ input\_marameter에 0xD3(211)값을 대입한다.
- ㉧ rcx 레지스터에 input\_parameter값을 저장한다.
- ㉨ for\_assembly\_function\_test 함수를 호출하며,  
함수 종료 시 돌아올(다음에 실행 될) 메모리 주소(return\_adress)를 저장한다.
- ㉩ for\_assembly\_function\_test 함수의 결과를 return\_value에 대입한다.



## for\_assembly\_function\_test

```

㉡ 00007FF66B441880  mov     dword ptr [rsp+8],ecx
㉢ 00007FF66B441884  push    rbp
㉣ 00007FF66B441885  push    rdi
㉤ 00007FF66B441886  sub     rsp,0E8h
㉥ 00007FF66B44188D  lea     rbp,[rsp+20h]
                                return number * 3;
㉦ 00007FF66B44189E  imul    eax,dword ptr [number],3
㉧ 00007FF66B4418A5  lea     rsp,[rbp+0C8h]
㉨ 00007FF66B4418AC  pop     rdi
㉩ 00007FF66B4418AD  pop     rbp
㉪ 00007FF66B4418AE  ret

```

㉡ rcx레지스터에 저장된 값을 number로써 rsp로부터 0x8byte 위쪽에 저장한다.

㉢ 현재 rbp값(000000048D70F8C0)을 저장한다.

㉣ 현재 rdi(0)값을 저장한다.

㉤ rsp로부터 0xE8(232)byte만큼 Stack공간을 확보한다.

㉥ rsp로부터 0x20(32)byte만큼 위쪽에 rbp값을 새로 지정한다.

㉦ number에 저장된 값에 3을 곱한 값을 출력한다.

㉧ rbp로부터 0xC8(200)byte만큼 위쪽을 rsp로 지정한다.

㉨ 저장되어있던 rdi값을 호출하여 rdi로 지정한다.

㉩ 저장되어있던 rbp값을 호출하여 rbp로 지정한다.

㉪ 저장되어있던 return\_adress를 rip로하여 되돌아간다.