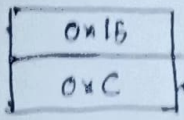
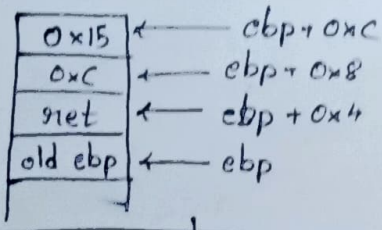
Q4) before calling

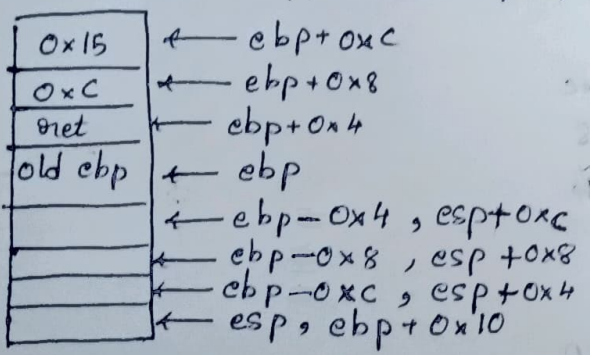


<+10> push ebp
↳ Pushing ebp to function stack

<+1> mov ebp, esp
making esp point to ebp



<+3> sub esp, 0x10 # esp=esp-0x10

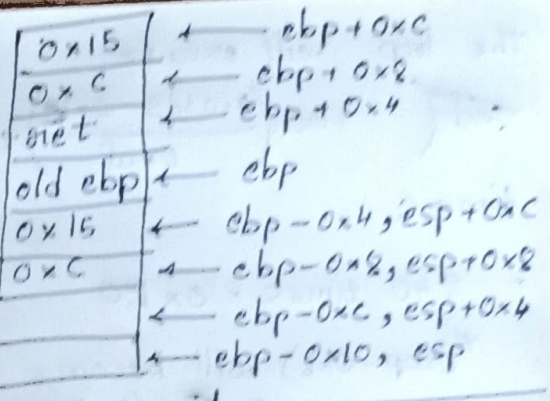


<+6> mov eax, DWORD PTR [ebp+0xc]
eax = 0x15

<+9> mov DWORD PTR [ebp-0x4], eax
*(ebp-0x4) = eax = 0x15

<+12> mov eax, DWORD PTR [ebp+0x8]
eax = *(ebp+0x8) = 0xc

<+15> mov DWORD PTR [ebp-0x8], eax
*(ebp-0x8) = eax = 0xc



<+18> jmp 0x50c <assemblycode +31>
jump to instruction at address 0x50c i.e assembly code +31

<+20> add DWORD PTR [ebp-0x4], 0x1
*(ebp-0x4) += 1

<+24> add DWORD PTR [ebp-0x8], 0xaf
*(ebp-0x8) += 175

<+31> cmp DWORD PTR [ebp-0x8], 0xa3d3

<+38> jle 0x501 <assemblycode +20>
if *(ebp-0x8) <= 41939 go to instruction on address assembly code +20

The above 5 lines are similar to:

```
while (*(ebp-0x8)<=41939){
    *(ebp-0x4) += 1;
    *(ebp-0x8) += 175;
}
```

The loop will execute

$$\text{floor}\left(\frac{0xa3d3 - 0xC}{0xaf}\right) + 1$$

times.

= 240 times = 0xF0

∴ *(ebp-0x8) will become

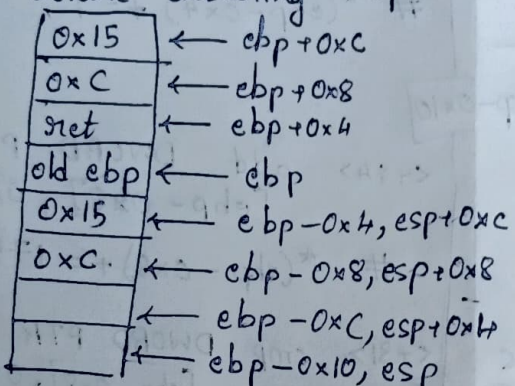
$$0xC + (0xF0 \cdot 0xaf) = 0xA41C$$

and *(ebp-0x4) will become

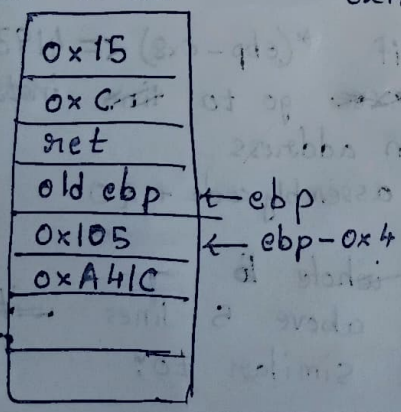
$$0x15 + (0xF0 \cdot 0x1)$$

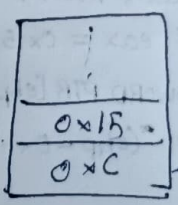
$$= 0x105$$

Stack before entering loop:



Stack after exiting loop.





Stack of Parent function

```
<+40>  mov  eax, DWORD PTR [ebp-0x4]
    # eax= *(ebp-0x4) = 0x105
<+43>  leave
    # make esp and ebp
    # as they were before
    # executing <+1>
<+44>  ret  #  Return.
```

5 (a)

- the given executable "q5.out" shows up in the directory and also on running the command "ls" in terminal.
- But on trying to execute it by running "./q5.out" bash throws the error "No such file or directory"

- The problem here is the given executable is a DYN (shared object file), but the interpreter was not correct one.

(b)

- the type of the ELF file DYN (shared object file)

- Machine - Advanced Micro Devices (x86-64)
- Data : 2's complement, Little endian