2. C 서브루틴 코드 분석

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

main 함수부터 시작

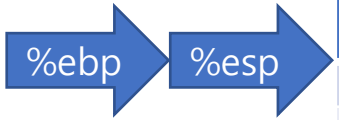
%ebp → %esp

STACK	
-4	
-8	
-12	
-16	
-20	
-24	
-28	
-32	
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	
ebx	
ecx	
edx	
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

스택 포인터, 베이스 포인터 초기화



STACK	
-4	
-8	
-12	
-16	
-20	
-24	
-28	
-32	
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	
ebx	
ecx	
edx	
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

로컬 변수를 위한 공간 확보

%ebp

%esp

STACK	
-4	
-8	
-12	
-16	
-20	
-24	
-28	
-32	
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	
ebx	
ecx	
edx	
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

___main 은 비어있음

%ebp

%esp

STACK	
-4	
-8	
-12	
-16	
-20	
-24	
-28	
-32	
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	
ebx	
ecx	
edx	
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

```
int a = 10;
```

%ebp

%esp

STACK	
-4	10
-8	
-12	
-16	
-20	
-24	
-28	
-32	
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	
ebx	
ecx	
edx	
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

```
int b = 10;
```

%ebp

%esp

STACK	
-4	10
-8	10
-12	
-16	
-20	
-24	
-28	
-32	
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	
ebx	
ecx	
edx	
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

int c = 5;

%ebp

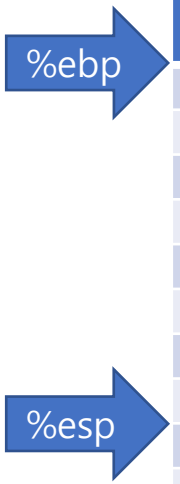
%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	
-32	
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	
ebx	
ecx	
edx	
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

add 함수를 위한 인자 세팅



STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	
-32	
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	5
ebx	
ecx	
edx	
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

add 함수를 위한 인자 세팅

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	5
ebx	
ecx	
edx	
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

add 함수를 위한 인자 세팅

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

add 함수를 위한 인자 세팅

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

add(b, c) 수행 시작

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	
eex	

```
                .globl  _add
                .def    _add; .scl 2; .type 32; .endef
_add:
LFB0:
                .cfi_startproc
                pushl   %ebp
                .cfi_def_cfa_offset 8
                .cfi_offset 5, -8
                movl    %esp, %ebp
                .cfi_def_cfa_register 5
                subl    $16, %esp
                movl    8(%ebp), %edx
                movl    12(%ebp), %eax
                addl    %edx, %eax
                movl    %eax, -4(%ebp)
                movl    -4(%ebp), %eax
                leave
                .cfi_restore 5
                .cfi_def_cfa 4, 4
                ret
                .cfi_endproc
LFE0:
```

add(b, c) 수행 시작

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	
eex	

```
                .globl  _add
                .def    _add; .scl 2; .type 32; .endef
_add:
LFB0:
                .cfi_startproc
                pushl   %ebp
                .cfi_def_cfa_offset 8
                .cfi_offset 5, -8
                movl    %esp, %ebp
                .cfi_def_cfa_register 5
                subl    $16, %esp
                movl    8(%ebp), %edx
                movl    12(%ebp), %eax
                addl    %edx, %eax
                movl    %eax, -4(%ebp)
                movl    -4(%ebp), %eax
                leave
                .cfi_restore 5
                .cfi_def_cfa 4, 4
                ret
                .cfi_endproc
LFE0:
```

이전 base pointer 위치 저장

%ebp

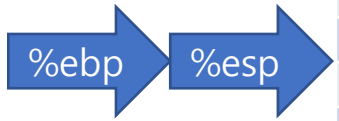
%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	10
-36	(former base pointer)
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	
eex	

```
        .globl  _add
        .def    _add; .scl 2; .type 32; .endef
_add:
LFB0:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        subl    $16, %esp
        movl    8(%ebp), %edx
        movl    12(%ebp), %eax
        addl    %edx, %eax
        movl    %eax, -4(%ebp)
        movl    -4(%ebp), %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE0:
```

Base pointer 끌어오기



STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	10
-36	(former base pointer)
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	
eex	

```
        .globl  _add
        .def    _add; .scl 2; .type 32; .endef
_add:
LFB0:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        subl    $16, %esp
        movl    8(%ebp), %edx
        movl    12(%ebp), %eax
        addl    %edx, %eax
        movl    %eax, -4(%ebp)
        movl    -4(%ebp), %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE0:
```

함수 내 로컬 변수 공간 확보

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	10
-36	(former base pointer)
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	
eex	

```
        .globl  _add
        .def    _add; .scl 2; .type 32; .endef
_add:
LFB0:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        subl    $16, %esp
        movl    8(%ebp), %edx
        movl    12(%ebp), %eax
        addl    %edx, %eax
        movl    %eax, -4(%ebp)
        movl    -4(%ebp), %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE0:
```

첫 번째 인자 가져오기

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	10
-36	(former base pointer)
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	10
eex	

```
        .globl  _add
        .def    _add; .scl 2; .type 32; .endef
_add:
LFB0:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        subl    $16, %esp
        movl    8(%ebp), %edx
        movl    12(%ebp), %eax
        addl    %edx, %eax
        movl    %eax, -4(%ebp)
        movl    -4(%ebp), %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE0:
```

두 번째 인자 가져오기

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	10
-36	(former base pointer)
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	5
ebx	
ecx	
edx	10
eex	

```
        .globl  _add
        .def    _add; .scl 2; .type 32; .endef
_add:
LFB0:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        subl    $16, %esp
        movl    8(%ebp), %edx
        movl    12(%ebp), %eax
        addl    %edx, %eax
        movl    %eax, -4(%ebp)
        movl    -4(%ebp), %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE0:
```

두 인자의 값 더하기

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	10
-36	(former base pointer)
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	15
ebx	
ecx	
edx	10
eex	

```
        .globl  _add
        .def    _add; .scl 2; .type 32; .endef
_add:
LFB0:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        subl    $16, %esp
        movl    8(%ebp), %edx
        movl    12(%ebp), %eax
        addl    %edx, %eax
        movl    %eax, -4(%ebp)
        movl    -4(%ebp), %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE0:
```

계산 결과 스택에 가져오기

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	10
-36	(former base pointer)
-40	
-44	15
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	15
ebx	
ecx	
edx	10
eex	

```
        .globl  _add
        .def    _add; .scl 2; .type 32; .endef
_add:
LFB0:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        subl    $16, %esp
        movl    8(%ebp), %edx
        movl    12(%ebp), %eax
        addl    %edx, %eax
        movl    %eax, -4(%ebp)
        movl    -4(%ebp), %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE0:
```

반환 값 레지스터에 저장하기

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	10
-36	(former base pointer)
-40	
-44	15
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	15
ebx	
ecx	
edx	10
eex	

```
        .globl  _add
        .def    _add; .scl 2; .type 32; .endef
_add:
LFB0:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        subl    $16, %esp
        movl    8(%ebp), %edx
        movl    12(%ebp), %eax
        addl    %edx, %eax
        movl    %eax, -4(%ebp)
        movl    -4(%ebp), %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE0:
```

베이스 포인터 회수

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	15
ebx	
ecx	
edx	10
eex	

```
                .globl  _add
                .def    _add; .scl 2; .type 32; .endef
_add:
LFB0:
                .cfi_startproc
                pushl   %ebp
                .cfi_def_cfa_offset 8
                .cfi_offset 5, -8
                movl    %esp, %ebp
                .cfi_def_cfa_register 5
                subl    $16, %esp
                movl    8(%ebp), %edx
                movl    12(%ebp), %eax
                addl    %edx, %eax
                movl    %eax, -4(%ebp)
                movl    -4(%ebp), %eax
                leave
                .cfi_restore 5
                .cfi_def_cfa 4, 4
                ret
                .cfi_endproc
LFE0:
```

함수 실행 종료

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	5
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	15
ebx	
ecx	
edx	10
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

함수 수행 결과 가져오기

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	15
ebx	
ecx	
edx	10
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

mul 실행을 위한 인자 세팅

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	10
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

mul 실행을 위한 인자 세팅

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	10
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

mul 수행 시작

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	10
eex	

```
                .globl  _mul
                .def    _mul;   .scl
                2;      .type   32;
                .endef
_mul:
LFB1:
                .cfi_startproc
                pushl   %ebp
                .cfi_def_cfa_offset 8
                .cfi_offset 5, -8
                movl    %esp, %ebp
                .cfi_def_cfa_register 5
                movl    8(%ebp), %eax
                imull   12(%ebp), %eax
                popl    %ebp
                .cfi_restore 5
                .cfi_def_cfa 4, 4
                ret
                .cfi_endproc
LFE1:
```

mul 수행 시작

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	10
eex	

```
        .globl  _mul
        .def    _mul;   .scl    2;      .type   32;     .endef
_mul:
LFB1:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        movl    8(%ebp), %eax
        imull   12(%ebp), %eax
        popl    %ebp
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE1:
```

이전 base pointer 위치 저장

%ebp

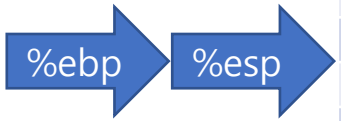
%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	(former base pointer)
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	10
eex	

```
        .globl  _mul
        .def    _mul;   .scl    2;      .type   32;     .endef
_mul:
LFB1:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        movl    8(%ebp), %eax
        imull   12(%ebp), %eax
        popl    %ebp
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE1:
```

Base pointer 끌어오기

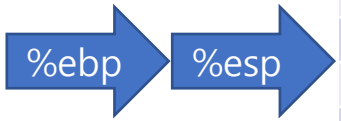


STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	(former base pointer)
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	10
eex	

```
        .globl  _mul
        .def    _mul;   .scl    2;      .type   32;     .endef
_mul:
LFB1:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        movl    8(%ebp), %eax
        imull   12(%ebp), %eax
        popl    %ebp
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE1:
```

함수 인자 가져오기

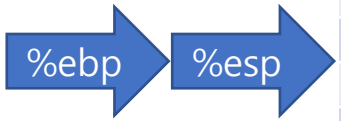


STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	(former base pointer)
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	10
ebx	
ecx	
edx	10
eex	

```
        .globl  _mul
        .def    _mul;   .scl    2;      .type   32;     .endef
_mul:
LFB1:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        movl    8(%ebp), %eax
        imull   12(%ebp), %eax
        popl    %ebp
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE1:
```

곱 연산 수행



STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	(former base pointer)
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	120
ebx	
ecx	
edx	10
eex	

```
        .globl  _mul
        .def    _mul;   .scl    2;      .type   32;     .endef
_mul:
LFB1:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        movl    8(%ebp), %eax
        imull   12(%ebp), %eax
        popl    %ebp
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE1:
```

Base pointer 회수

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	120
ebx	
ecx	
edx	10
eex	

```
        .globl  _mul
        .def    _mul;   .scl    2;      .type   32;     .endef
_mul:
LFB1:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        movl    8(%ebp), %eax
        imull   12(%ebp), %eax
        popl    %ebp
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE1:
```

함수 실행 종료

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	120
ebx	
ecx	
edx	10
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

int *ret 주소 가져오기

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	120
ebx	
ecx	
edx	(address)
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

함수 수행 결과를 ret의 주소에 저장

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	120
ebx	
ecx	
edx	(address)
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

main 함수의 반환값으로 0 설정

%ebp

%esp

STACK	
-4	10
-8	10
-12	5
-16	
-20	
-24	
-28	15
-32	10
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	0
ebx	
ecx	
edx	(address)
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

Base pointer 회수

%ebp %esp

STACK	
-4	
-8	
-12	
-16	
-20	
-24	
-28	
-32	
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	0
ebx	
ecx	
edx	(address)
eex	

```
        .globl  _main
        .def    _main; .scl 2; .type 32; .endef
_main:
LFB2:
        .cfi_startproc
        pushl   %ebp
        .cfi_def_cfa_offset 8
        .cfi_offset 5, -8
        movl    %esp, %ebp
        .cfi_def_cfa_register 5
        andl    $-16, %esp
        subl    $32, %esp
        call    ___main
        movl    $10, 28(%esp)
        movl    $10, 24(%esp)
        movl    $5, 20(%esp)
        movl    20(%esp), %eax
        movl    %eax, 4(%esp)
        movl    24(%esp), %eax
        movl    %eax, (%esp)
        call    _add
        movl    %eax, 4(%esp)
        movl    28(%esp), %eax
        movl    %eax, (%esp)
        call    _mul
        movl    16(%esp), %edx
        movl    %eax, (%edx)
        movl    $0, %eax
        leave
        .cfi_restore 5
        .cfi_def_cfa 4, 4
        ret
        .cfi_endproc
LFE2:
```

main 함수 실행 종료

%ebp

%esp

STACK	
-4	
-8	
-12	
-16	
-20	
-24	
-28	
-32	
-36	
-40	
-44	
-48	
-52	
-56	
-60	
-64	
-68	
-72	
-76	
-80	
-84	
-88	
-92	
-96	
-100	

REGISTERS	
eax	0
ebx	
ecx	
edx	(address)
eex	