Returning Value From Function

Function return value can be either returned in a register or on the stack. For all objects of basic data types (int, char, pointer etc) is returned in eax register. The structure is returned on stack. We will see this is the next chapter.

C code:

```
int fun()
{
    return 16;
}
int main()
{
    int a;
    a = fun();
    return 0;
}
```

Generated assembly code:

```
    .text
.globl fun
fun:
    pushl   %ebp
    movl    %esp, %ebp
    movl    $16, %eax
    popl    %ebp
    ret
.globl main
main:
    pushl   %ebp
    movl    %esp, %ebp
    subl    $16, %esp
    call    fun
    movl    %eax, -4(%ebp)
    movl    $0, %eax
    leave
    ret
```

Comments on generated code:

```
# text segment starts here
    .text

# export the function fun
.globl fun

# code of the function fun starts from here
fun:

# save the ebp register and set the ebp register to point the current stack frame
    pushl   %ebp
    movl    %esp, %ebp

# move return value (16) to eax register
    movl    $16, %eax

# restore the ebp register and return.
# Note that the return instruction will not change the value
# of each register. The caller of this function will get the
# return value in eax register
    popl    %ebp
    ret

# export the main function.
.globl main

# code for function main starts from here
main:

# save the ebp register and point it to current stack frame
    pushl   %ebp
    movl    %esp, %ebp

# create local variable and make proper alignment of the esp register.
    subl    $16, %esp

# Call the function fun.
    call    fun
```

```
# The function fun has returned in eax register.
# Store this value in local variable a
    movl    %eax, -4(%ebp)

# the main function returns now. It will return value in the eax register.
    movl    $0, %eax
    leave
    ret
```

‹ Parameter Passing (/cin/parameter.html)
up (/cin/cin.html)
Memory Allocation for Structure › (/cin/structure.html)