

Pointer Dereferencing

Take this C code as example:

```
// this is defined globally.  
int globalVar;  
// These lines of code are part of some function  
int b;  
int * ptr;  
ptr = &globalVar;  
*ptr = 100;  
b = *ptr;
```

Generated assembly code:

```
movl  $globalVar, -4(%ebp)  
movl  -4(%ebp), %eax  
movl  $100, (%eax)  
movl  -4(%ebp), %eax  
movl  (%eax), %eax  
movl  %eax, -8(%ebp)
```

Location of local variables of the stack (local variables are explained here ([memorymanagement.html](#)))

```
ptr => -4(%ebp)  
b => -8(%ebp)
```

The use of registers as temporary memory is described here ([arithmeticop.html#tempVaribaleUsage](#))

Comments on generated assembly code:

```
# ptr = &globalVar
    movl    $globalVar, -4(%ebp)

# tmp = ptr
    movl    -4(%ebp), %eax

# *tmp = 100
    movl    $100, (%eax)

# tmp = ptr
    movl    -4(%ebp), %eax

# tmp = *tmp
    movl    (%eax), %eax

# b = tmp
    movl    %eax, -8(%ebp)
```

[◀ Translation of Loop \(/cin/loop.html\)](/cin/loop.html)

[up \(/cin/cin.html\)](/cin/cin.html)

[Array Indexing ▶ \(/cin/array.html\)](/cin/array.html)

Do you collaborate using whiteboard? Please try Lekh Board - An Intelligent Collaborate Whiteboard App (<https://lekh.app>)
