

汇编代码的执行流程

```
mov eax, 2  
mov [x], eax  
mov eax, 3  
mov [y], eax  
mov eax, [x]  
mov ebx, [y]  
add eax, ebx  
mov [z], eax  
mov eax, [z]  
ret
```

这里的四步被替换为了两步

将[x]赋值为2

将[y]赋值为3

将要保存在[x]加的值放入
寄存器eax准备运算

mov eax, 2

mov [x], eax

mov eax, 3

mov [y], eax

mov eax, [x]

eax ebx



eax ebx



eax ebx



eax ebx



eax ebx



x y z



x y z



x y z



x y z



x y z

将保存在[y]里的要加的值
放入寄存器ebx准备运算

将两个数加起来
放入eax中

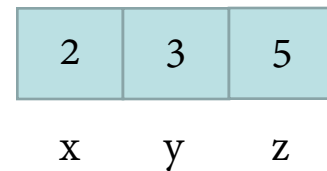
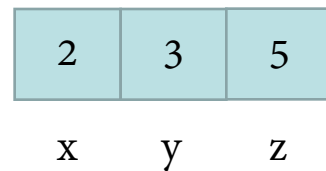
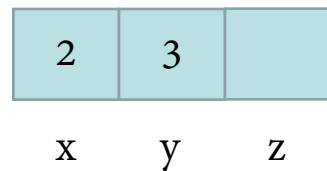
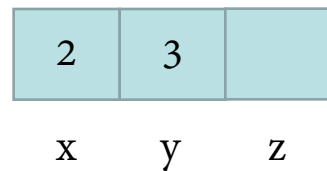
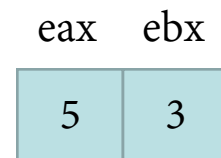
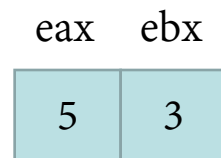
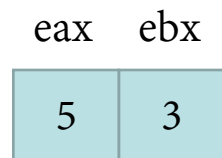
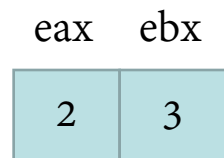
将结果取出又放回eax中

mov ebx, [y]

add eax, ebx

mov [z], eax

mov eax, [z]



C语言代码的执行流程

```
mov [x],0x2  
mov [y],0x3  
mov edx,[x]  
mov eax,[y]  
add eax,edx  
mov [z],eax  
mov eax,[z]  
ret
```

将要运算的值赋值到[x]和[y]

将要运算的值加载进寄存器

将两个寄存器中的值相加

mov [x], 2

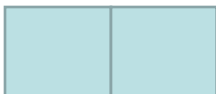
mov [y], 3

mov edx, [x]

mov eax, [y]

add eax, edx

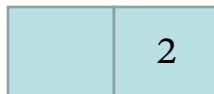
eax edx



eax edx



eax edx



eax ebx



eax edx



x y z



x y z



x y z



x y z



x y z

将结果重新加载到eax上

mov [z], eax

mov eax, [z]

eax ebx

5	2
---	---

eax ebx

5	
---	--

2	3	5
---	---	---

x y z

2	3	5
---	---	---

x y z