

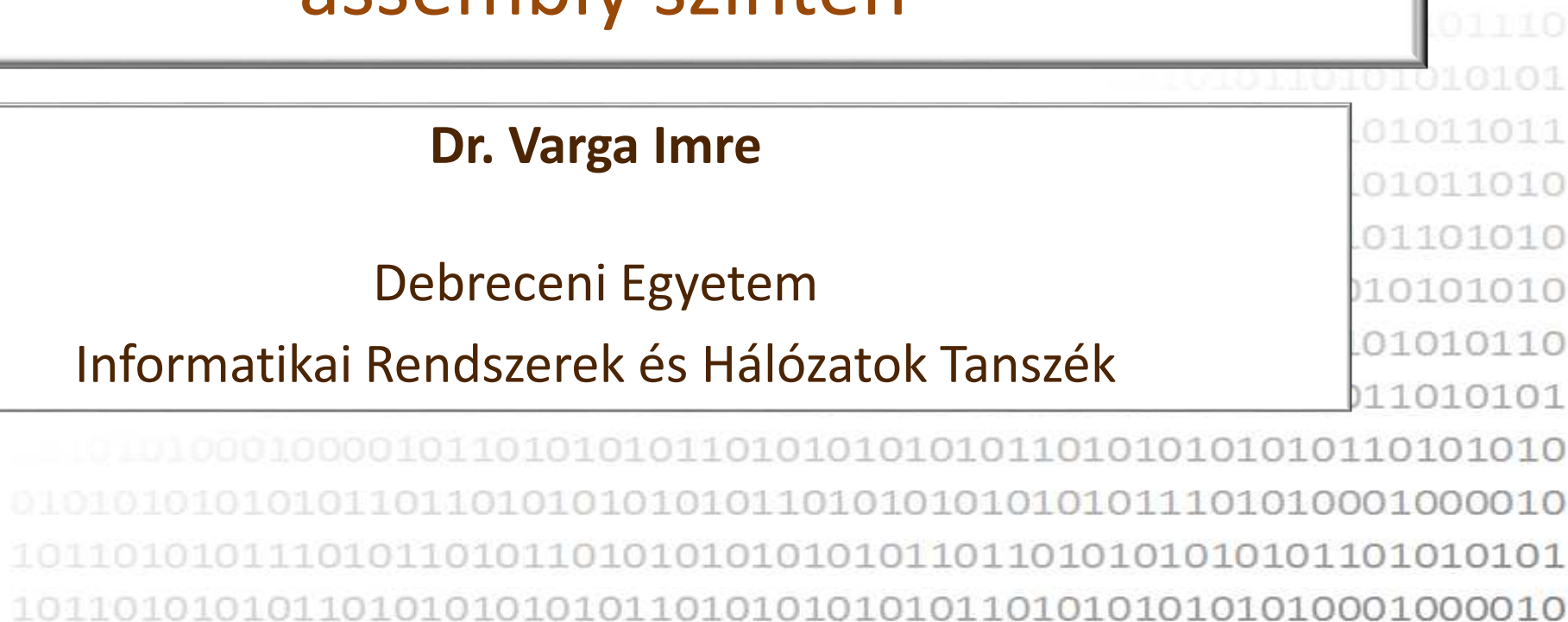
Függvényhívás assembly szinten

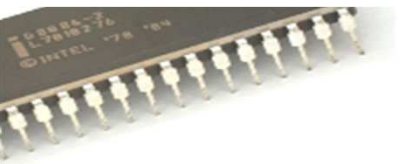
Dr. Varga Imre

Debreceni Egyetem

Informatikai Rendszerek és Hálózatok Tanszék

2018.01.29

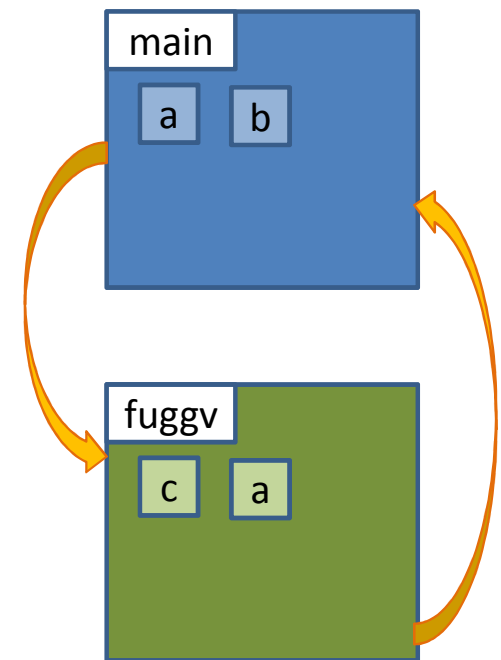


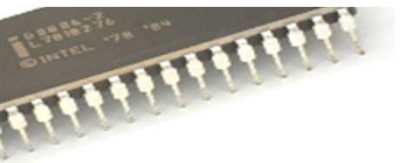


Példa C nyelven

```
int fuggv(int c) {  
    int a;  
    a=c+1;  
    return a;  
}  
int main(int argc, char *argv[]) {  
    int a, b;  
    a=argc;  
    b=fuggv(a);  
    return b;  
}
```

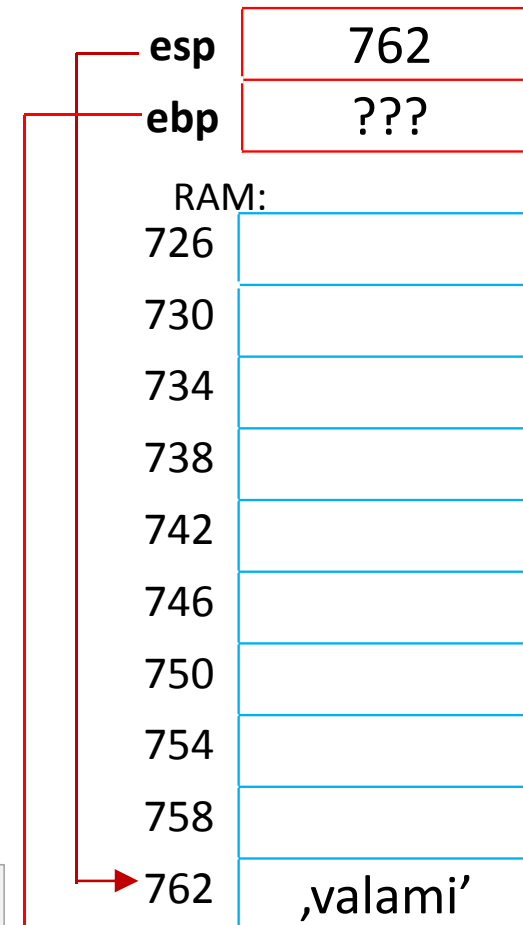
```
gcc prog.c -o prog
```



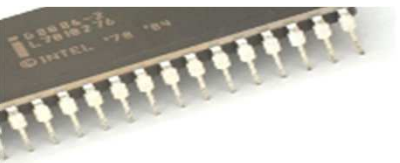


Példa assembly szinten

Kiindulási állapot programindításkor:



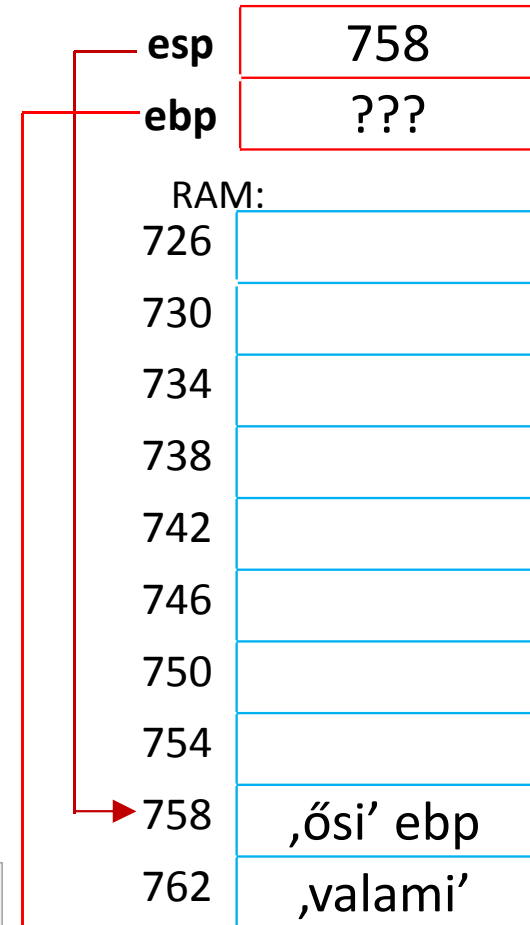
`./prog`

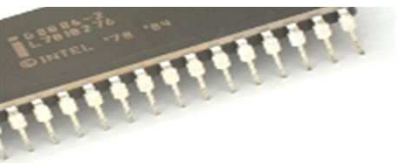


Példa assembly szinten

```
main:    push    ebp
```

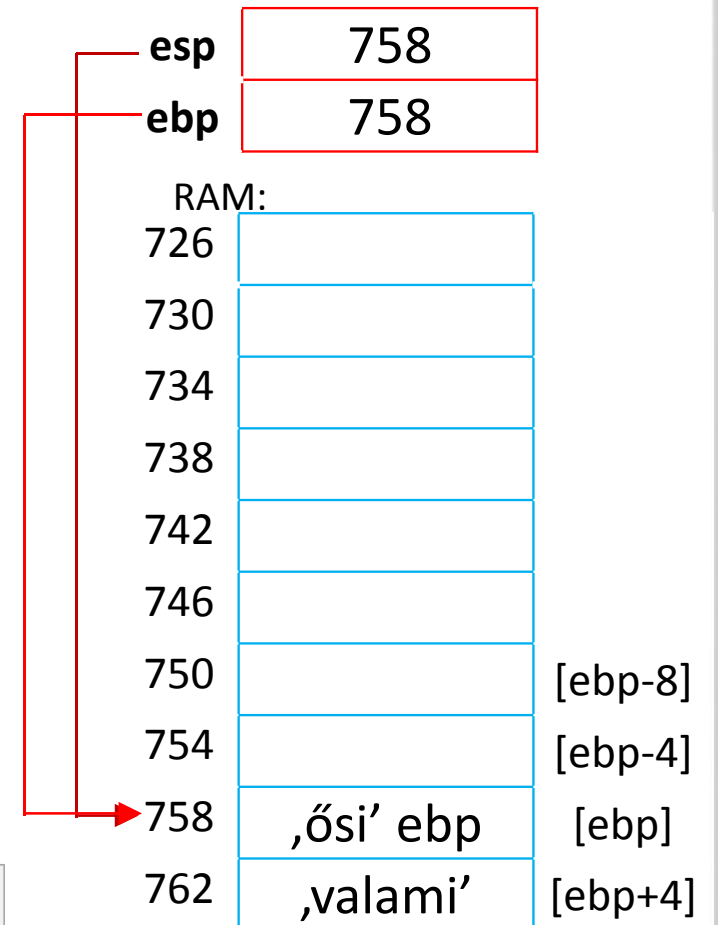
```
int main() {
```



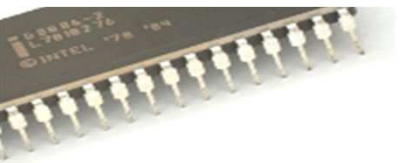


Példa assembly szinten

```
main:  push  ebp
       mov  ebp, esp
```



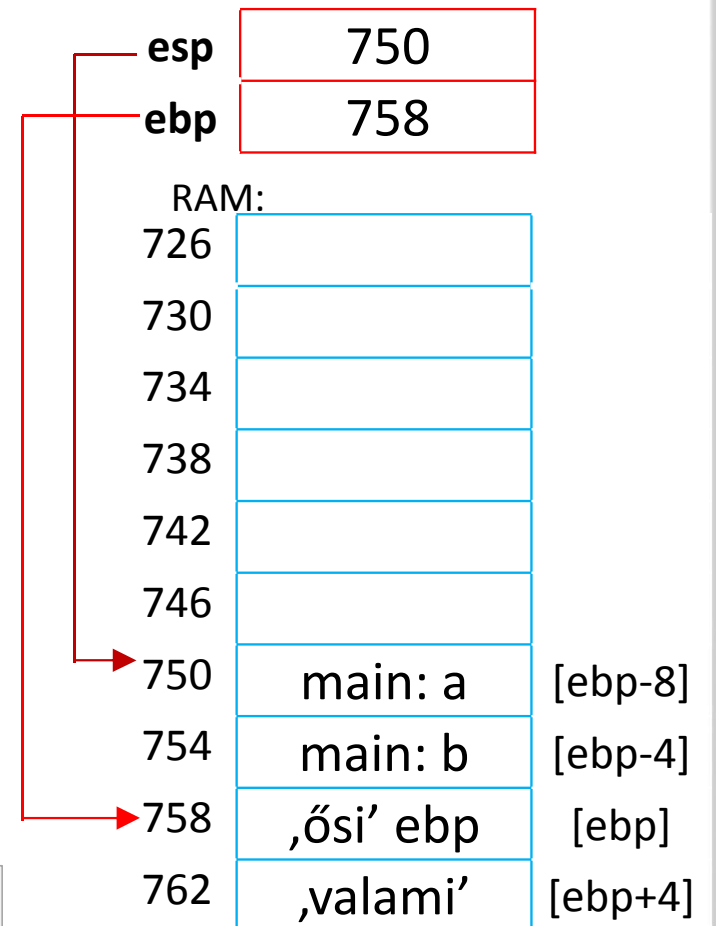
```
int main() {
```

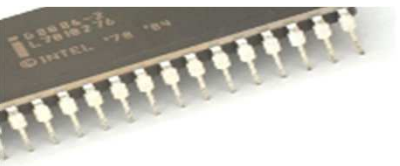


Példa assembly szinten

```
main:  push    ebp
       mov     ebp, esp
       sub     esp, 8
```

```
int a, b;
```

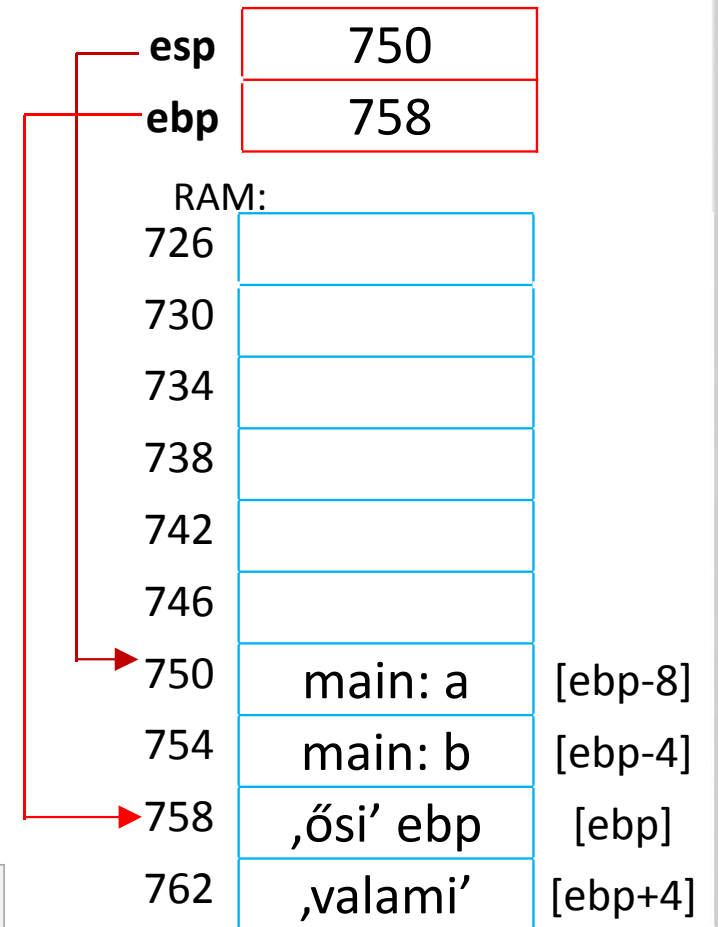


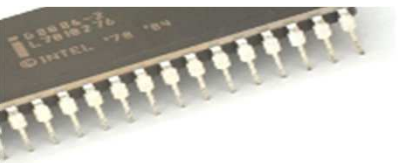


Példa assembly szinten

```
main:  push  ebp
       mov  ebp, esp
       sub  esp, 8
       mov  DWORD PTR [ebp-8], edi
```

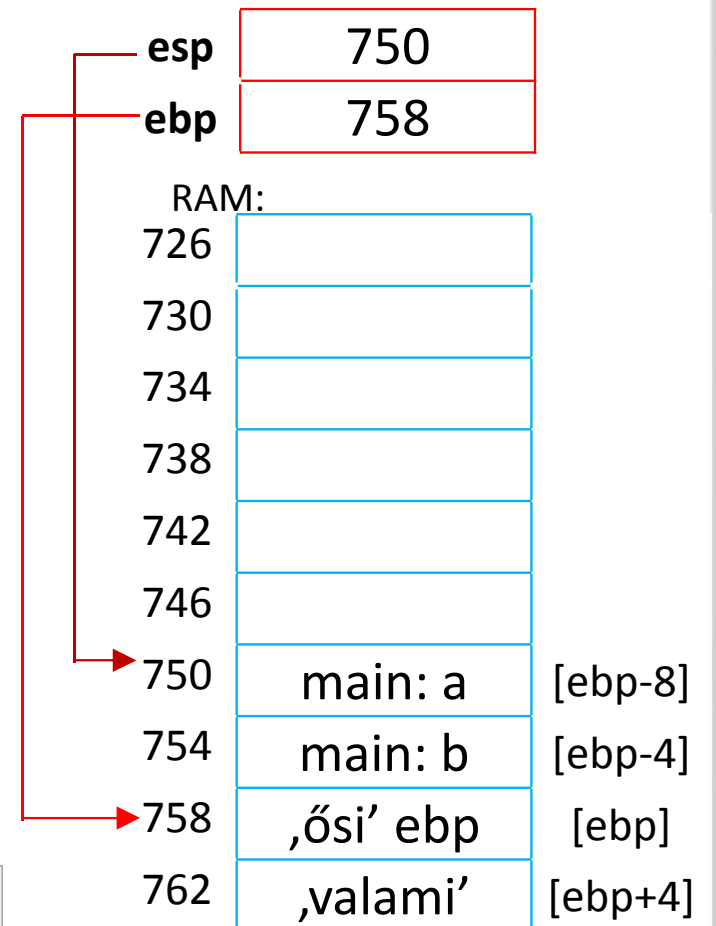
```
a=argc;
```



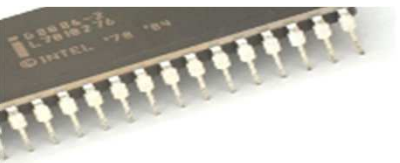


Példa assembly szinten

```
main:  push    ebp
       mov     ebp, esp
       sub     esp, 8
       mov     DWORD PTR [ebp-8], edi
       mov     eax, DWORD PTR [ebp-8]
```

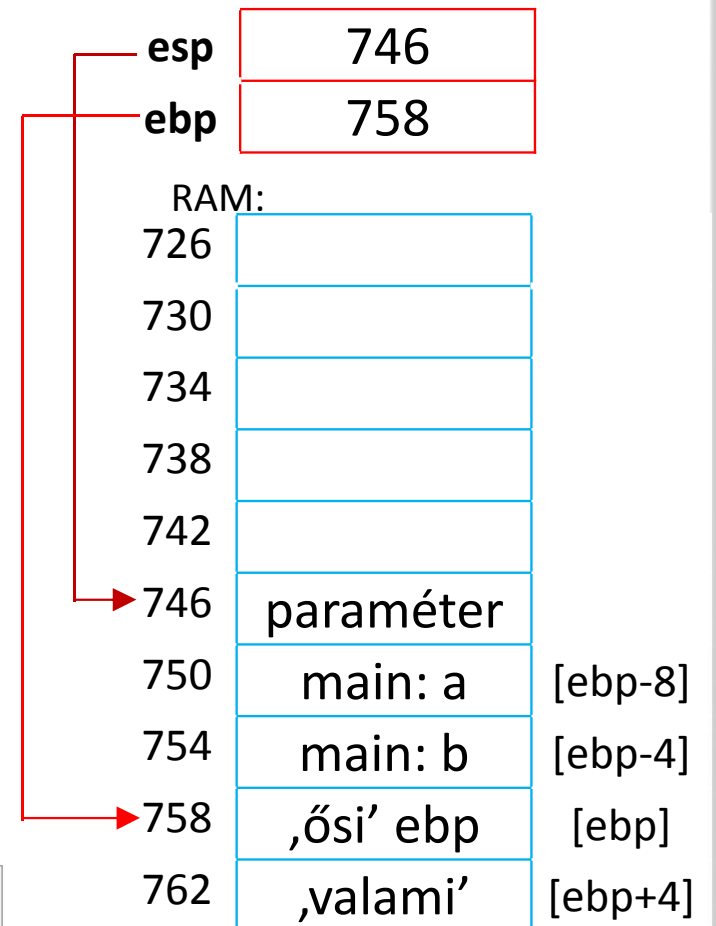


b=fuggv (a) ;

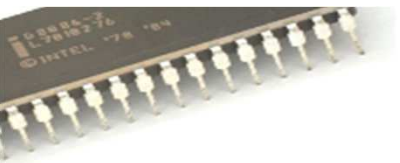


Példa assembly szinten

```
main:  push    ebp
       mov     ebp, esp
       sub     esp, 8
       mov     DWORD PTR [ebp-8], edi
       mov     eax, DWORD PTR [ebp-8]
       push   eax
```



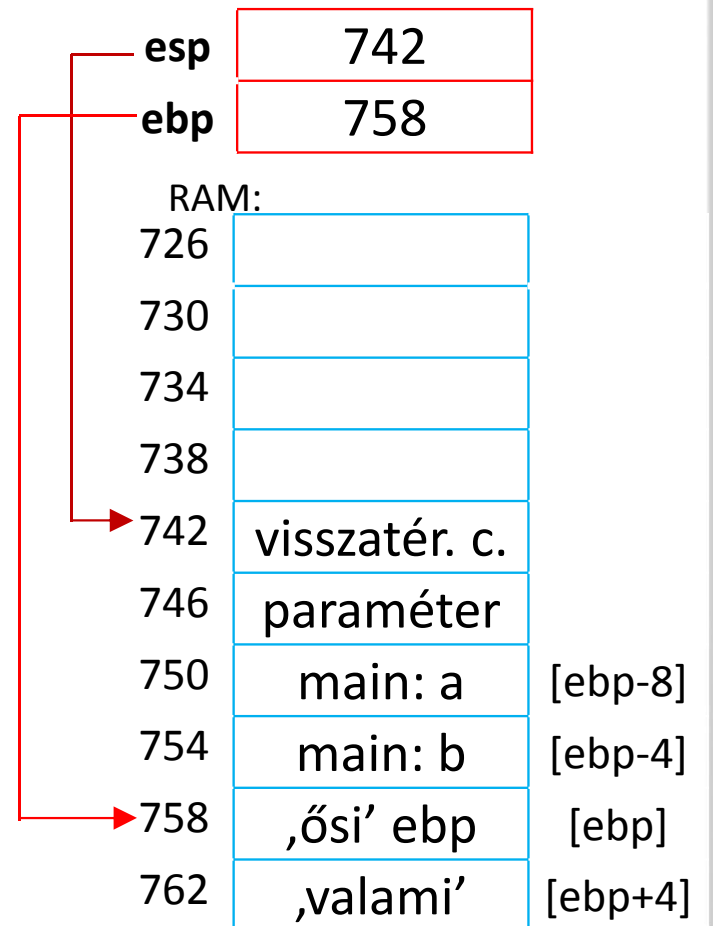
```
b=fuggv(a);
```

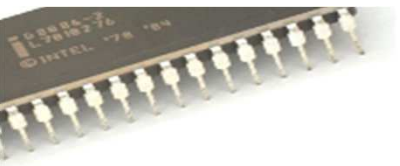


Példa assembly szinten

```
main:  push  ebp
       mov  ebp, esp
       sub  esp, 8
       mov  DWORD PTR [ebp-8], edi
       mov  eax, DWORD PTR [ebp-8]
       push eax
       call fuggv
```

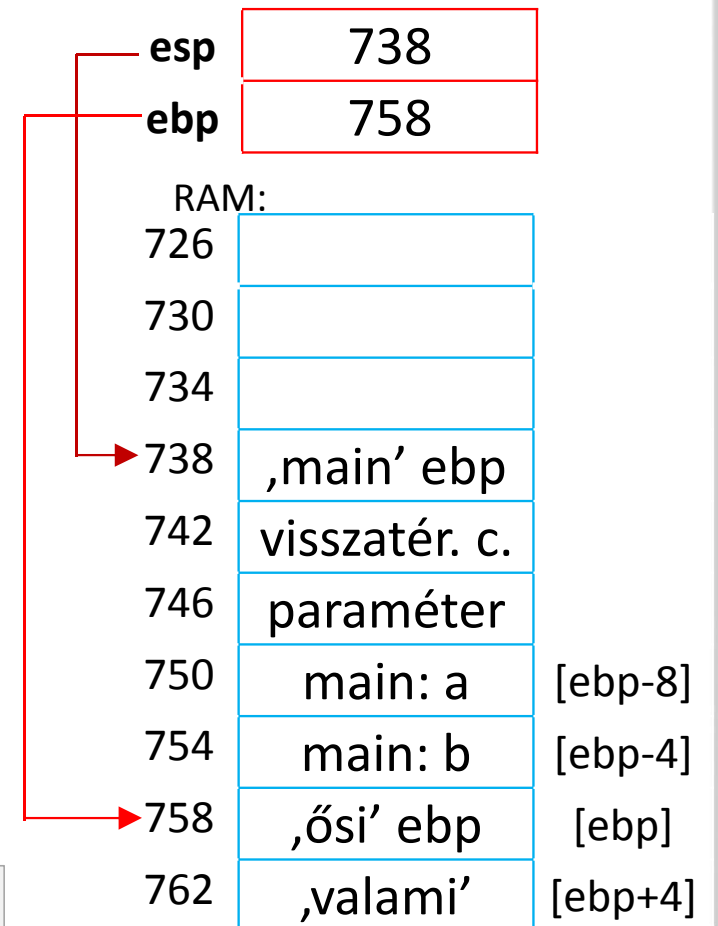
```
b=fuggv(a);
```



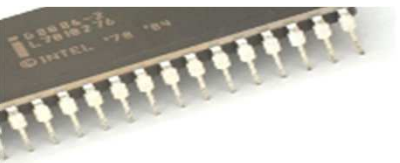


Példa assembly szinten

```
fuggv:  push  ebp
```

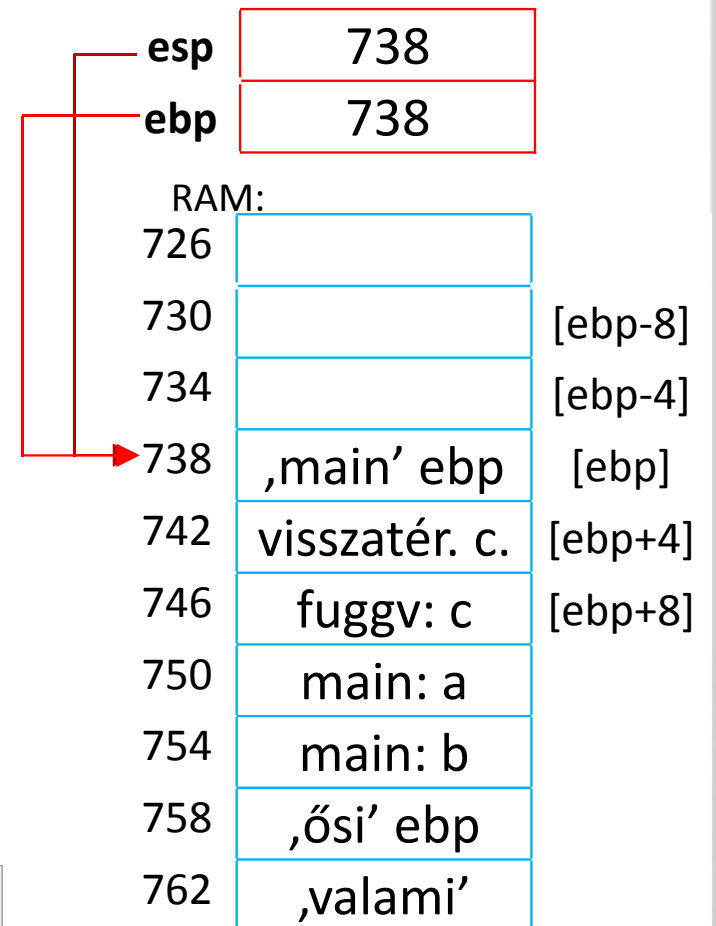


```
int fuggv(int c){
```

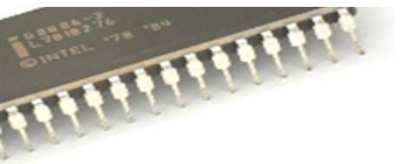


Példa assembly szinten

```
fuggv:  push  ebp
        mov   ebp, esp
```

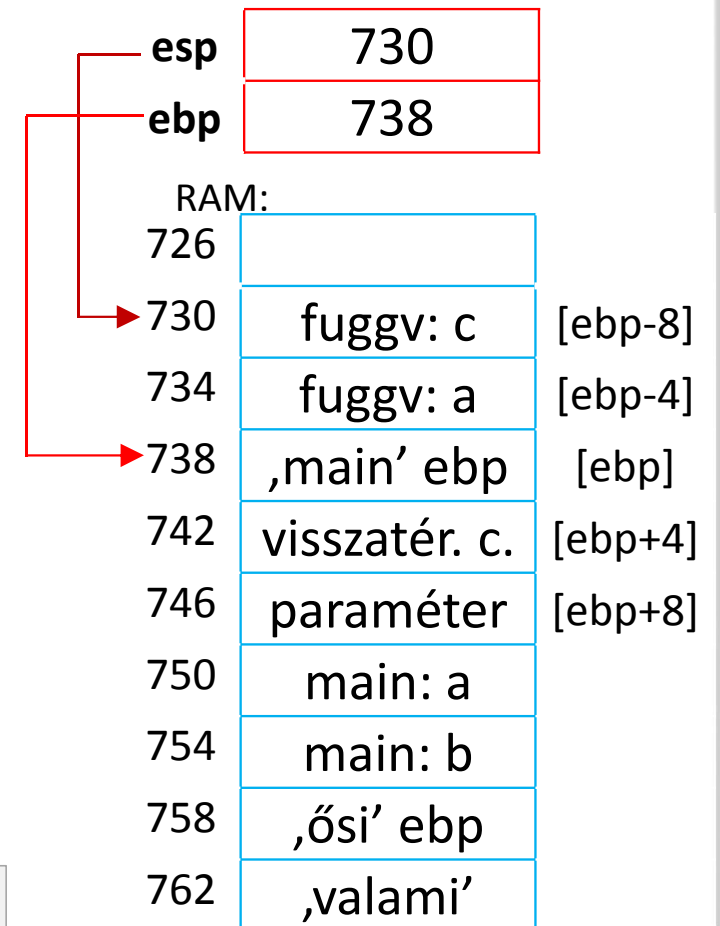


```
int fuggv(int c){
```

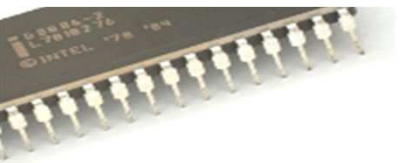


Példa assembly szinten

```
fuggv:  push    ebp
        mov     ebp, esp
        sub     esp, 8
```

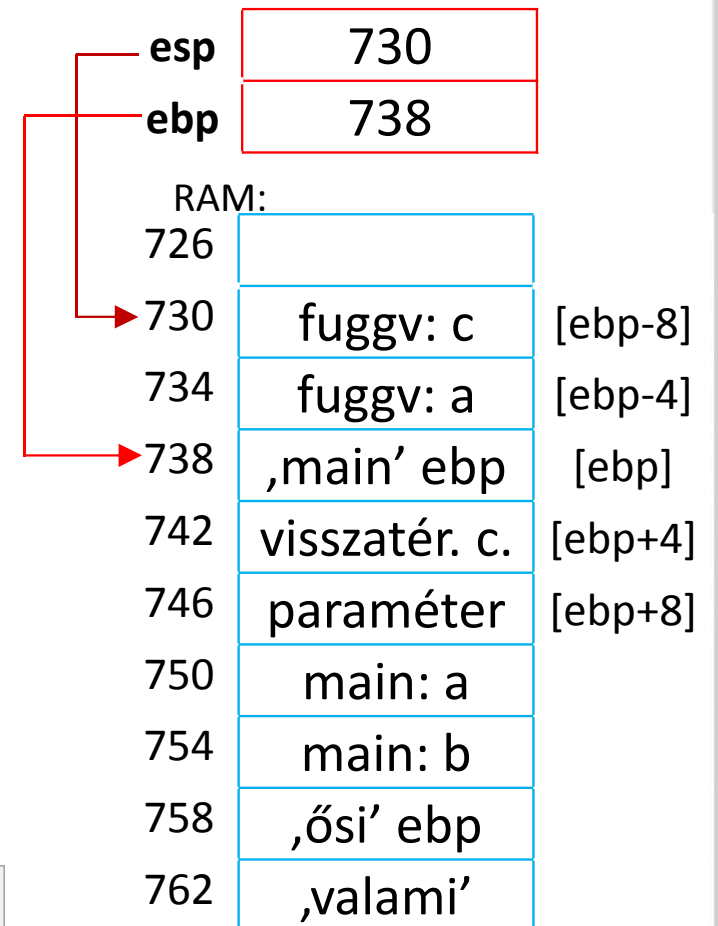


```
int fuggv(int c){ int a;
```

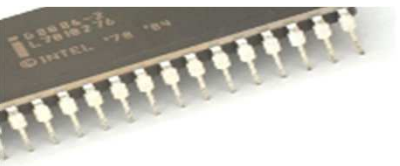


Példa assembly szinten

```
fuggv:  push    ebp
        mov     ebp, esp
        sub     esp, 8
        mov     eax, DWORD PTR [ebp+8]
```

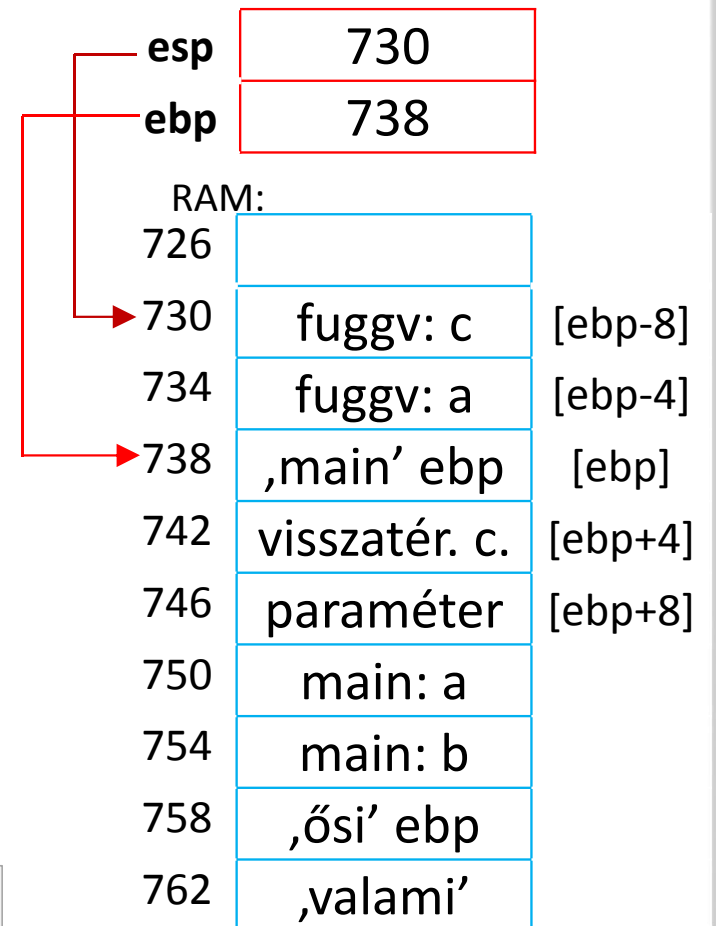


```
int fuggv(int c){
```

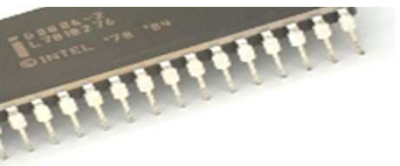


Példa assembly szinten

```
fuggv:  push    ebp
        mov     ebp, esp
        sub     esp, 8
        mov     eax, DWORD PTR [ebp+8]
        mov     DWORD PTR [ebp-8], eax
```

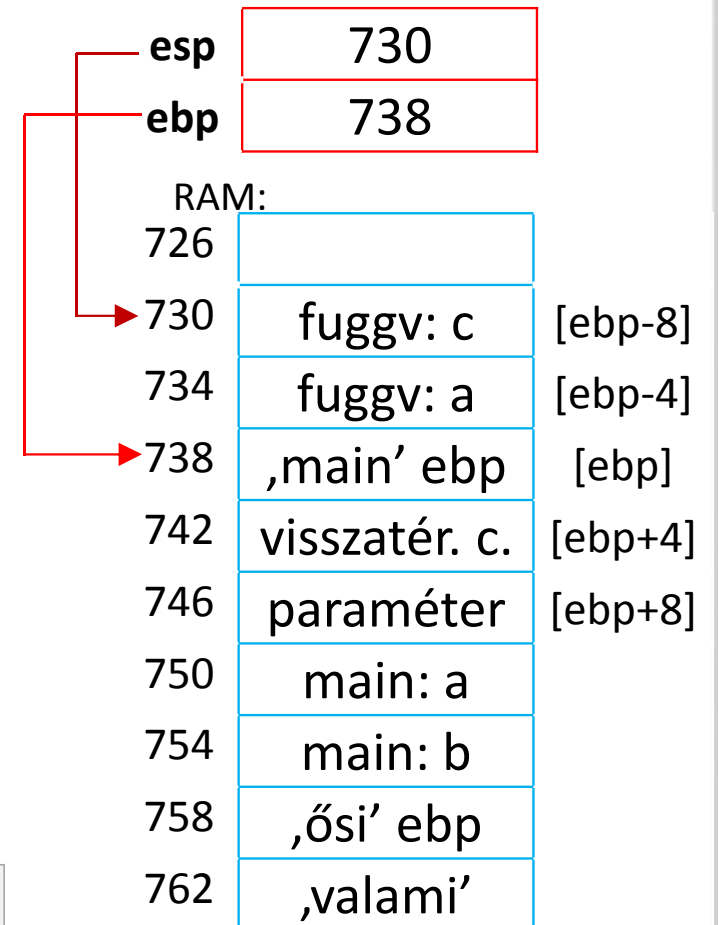


```
int fuggv(int c){
```

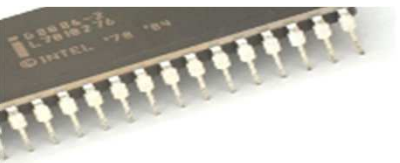


Példa assembly szinten

```
fuggv:  push    ebp
        mov     ebp, esp
        sub     esp, 8
        mov     eax, DWORD PTR [ebp+8]
        mov     DWORD PTR [ebp-8], eax
        mov     eax, DWORD PTR [ebp-8]
```



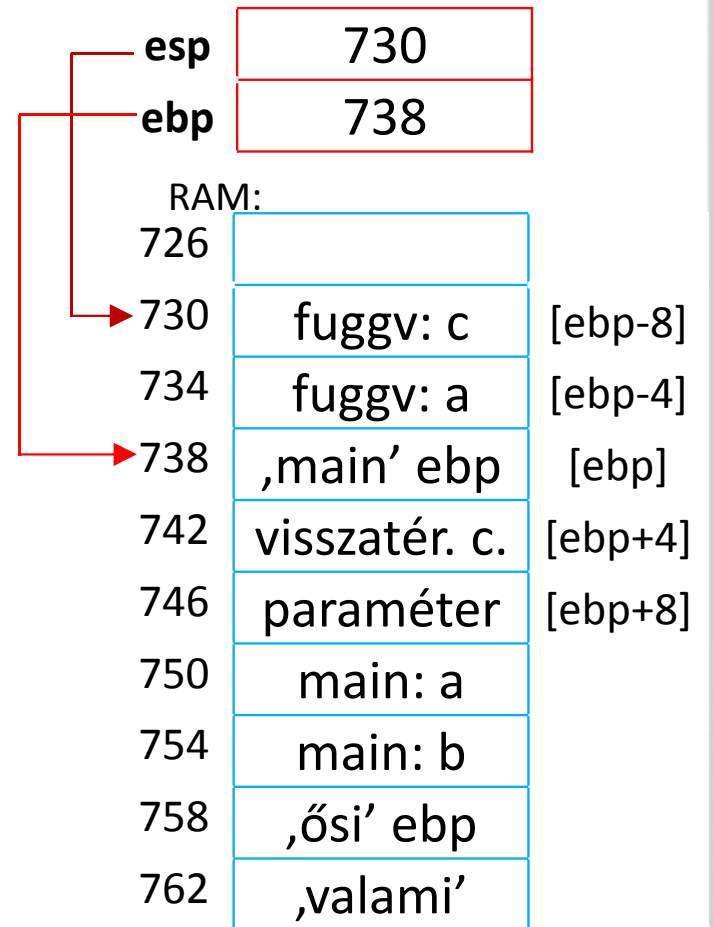
```
a=c+1;
```

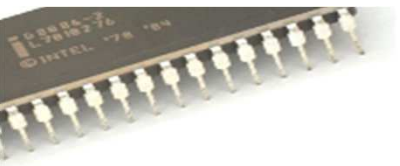



Példa assembly szinten

```
fuggv:  push    ebp
        mov     ebp, esp
        sub     esp, 8
        mov     eax, DWORD PTR [ebp+8]
        mov     DWORD PTR [ebp-8], eax
        mov     eax, DWORD PTR [ebp-8]
        add     eax, 1
```

```
a=c+1;
```

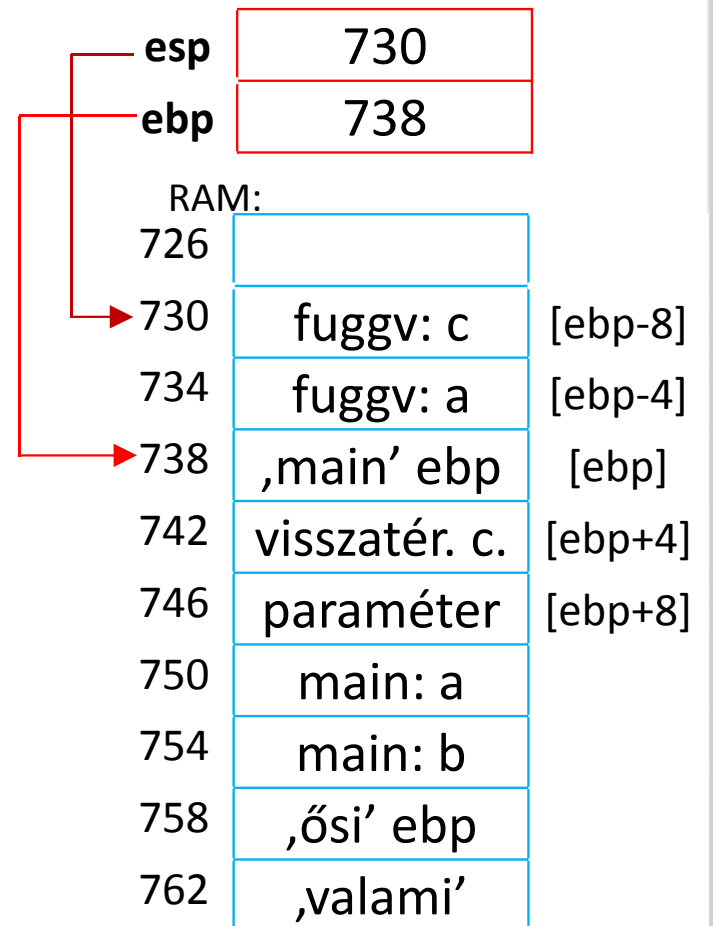


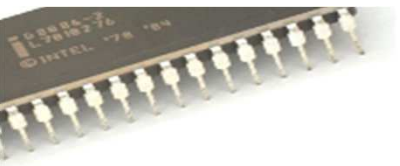


Példa assembly szinten

```
fuggv:  push    ebp
        mov     ebp, esp
        sub     esp, 8
        mov     eax, DWORD PTR [ebp+8]
        mov     DWORD PTR [ebp-8], eax
        mov     eax, DWORD PTR [ebp-8]
        add     eax, 1
        mov     DWORD PTR [ebp-4], eax
```

```
a=c+1;
```

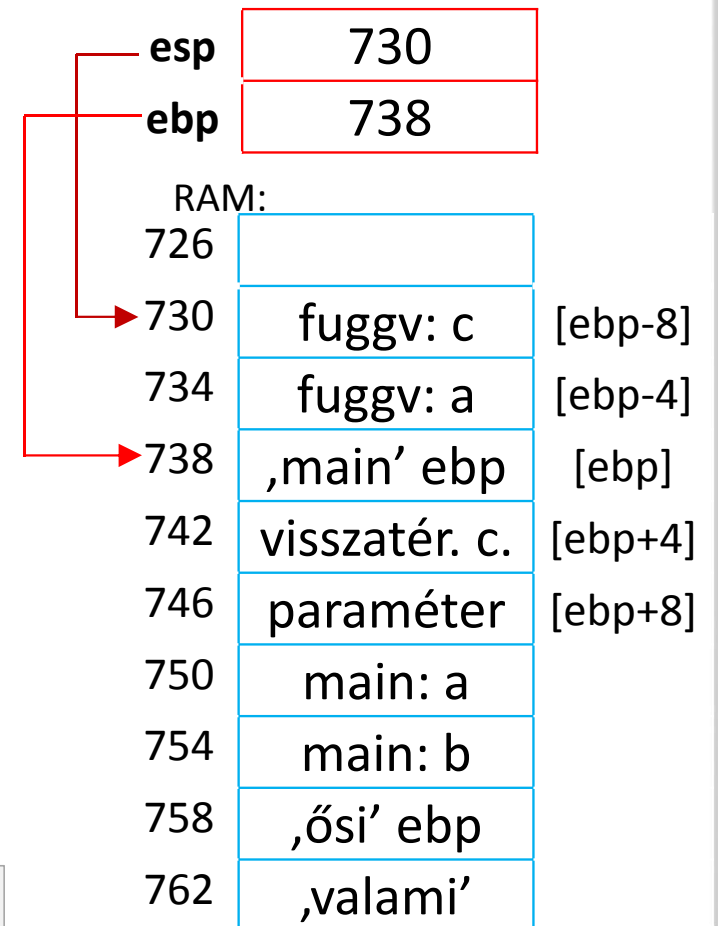


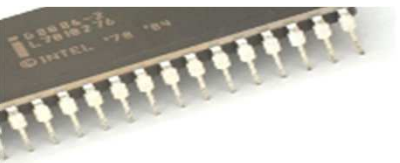


Példa assembly szinten

```
fuggv:  push    ebp
        mov     ebp, esp
        sub     esp, 8
        mov     eax, DWORD PTR [ebp+8]
        mov     DWORD PTR [ebp-8], eax
        mov     eax, DWORD PTR [ebp-8]
        add     eax, 1
        mov     DWORD PTR [ebp-4], eax
        mov     eax, DWORD PTR [ebp-4]
```

```
return a;
```

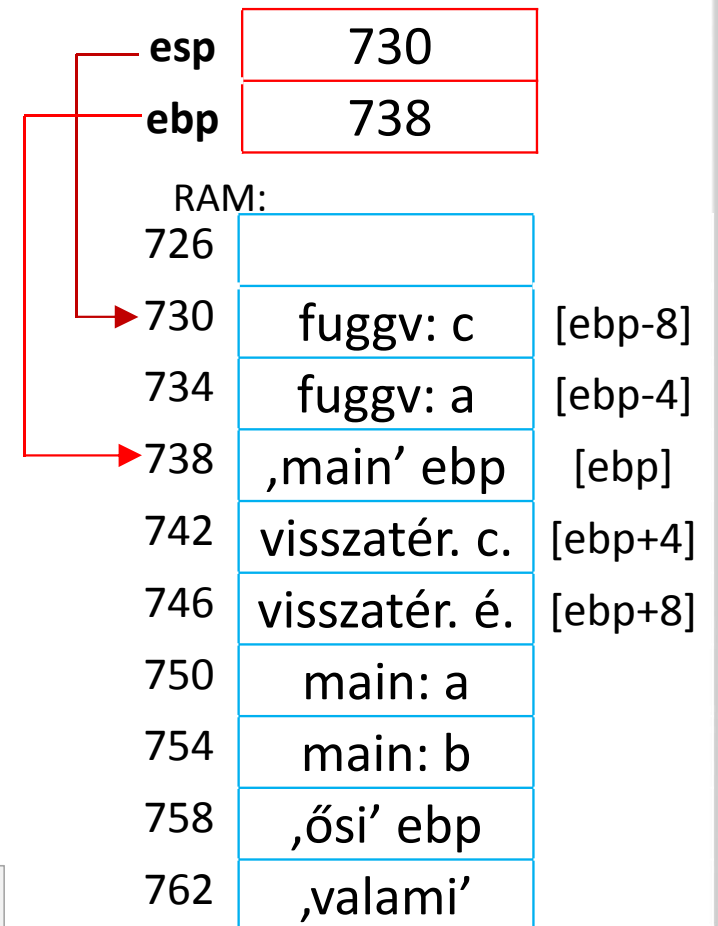


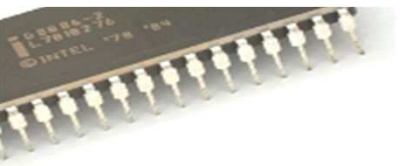


Példa assembly szinten

```
fuggv:  push    ebp
        mov     ebp, esp
        sub     esp, 8
        mov     eax, DWORD PTR [ebp+8]
        mov     DWORD PTR [ebp-8], eax
        mov     eax, DWORD PTR [ebp-8]
        add     eax, 1
        mov     DWORD PTR [ebp-4], eax
        mov     eax, DWORD PTR [ebp-4]
        mov     DWORD PTR [ebp+8], eax
```

```
return a;
```

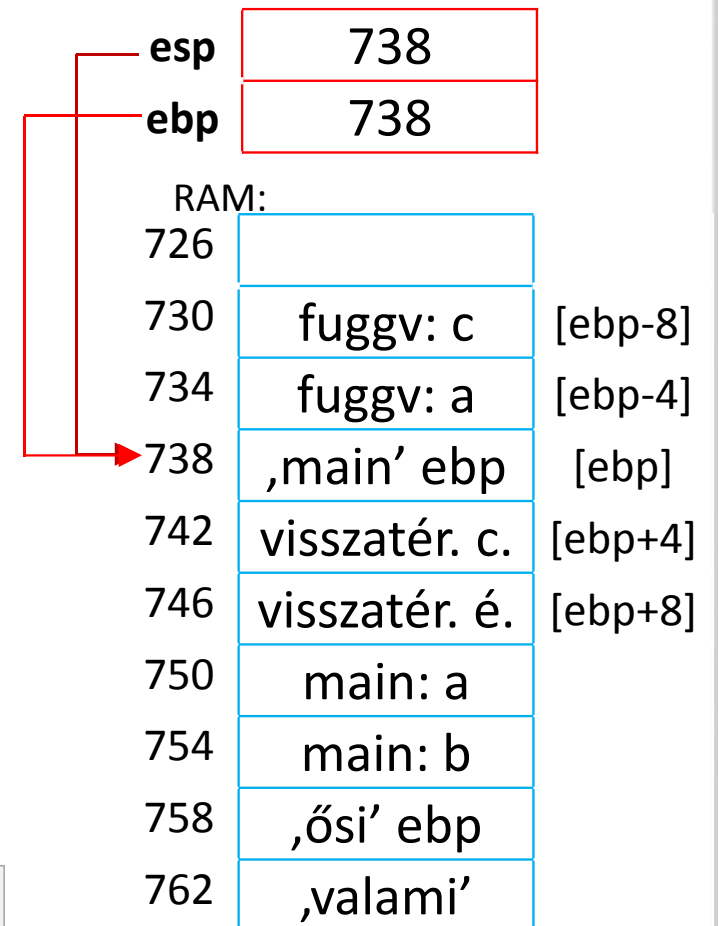


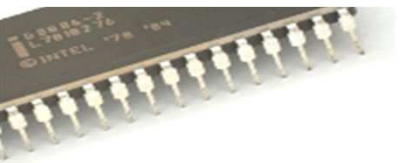


Példa assembly szinten

```
fuggv:  push    ebp
        mov     ebp, esp
        sub     esp, 8
        mov     eax, DWORD PTR [ebp+8]
        mov     DWORD PTR [ebp-8], eax
        mov     eax, DWORD PTR [ebp-8]
        add     eax, 1
        mov     DWORD PTR [ebp-4], eax
        mov     eax, DWORD PTR [ebp-4]
        mov     DWORD PTR [ebp+8], eax
        mov     esp, ebp
```

```
return a;
```

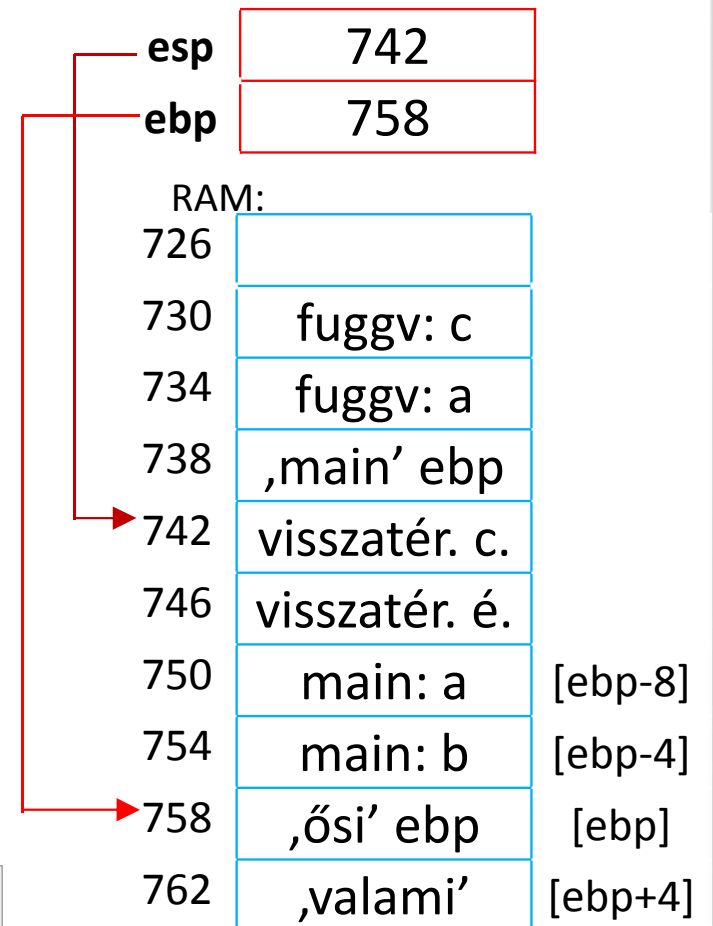


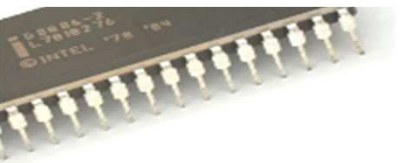


Példa assembly szinten

```
fuggv:  push    ebp
        mov     ebp, esp
        sub     esp, 8
        mov     eax, DWORD PTR [ebp+8]
        mov     DWORD PTR [ebp-8], eax
        mov     eax, DWORD PTR [ebp-8]
        add     eax, 1
        mov     DWORD PTR [ebp-4], eax
        mov     eax, DWORD PTR [ebp-4]
        mov     DWORD PTR [ebp+8], eax
        mov     esp, ebp
        pop     ebp
```

```
return a;
```

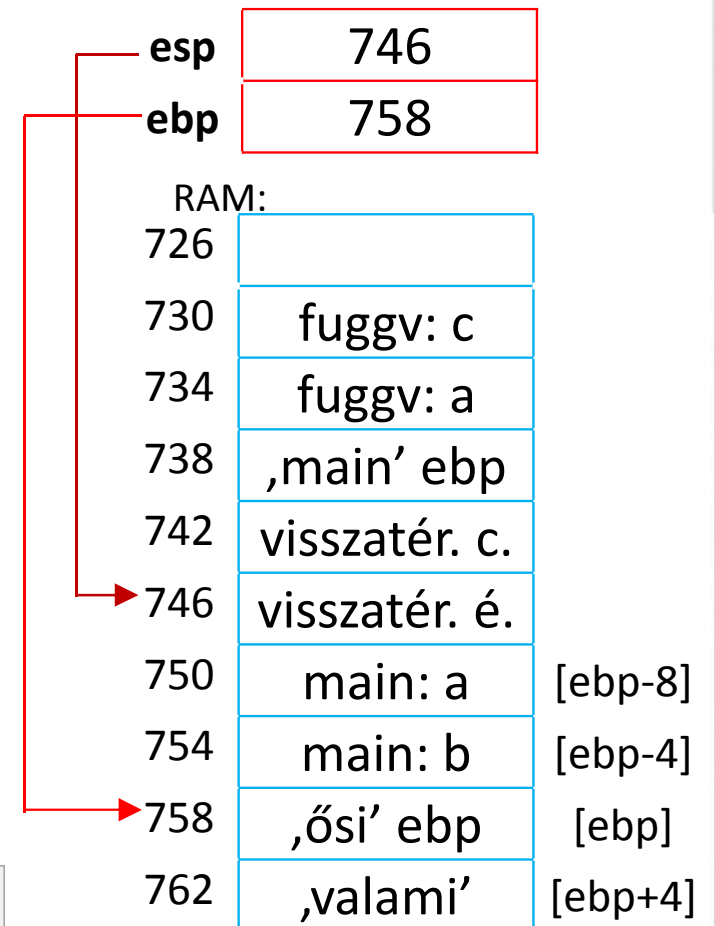


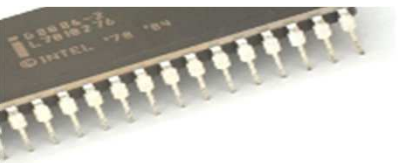


Példa assembly szinten

```
fuggv:  push    ebp
        mov     ebp, esp
        sub     esp, 8
        mov     eax, DWORD PTR [ebp+8]
        mov     DWORD PTR [ebp-8], eax
        mov     eax, DWORD PTR [ebp-8]
        add     eax, 1
        mov     DWORD PTR [ebp-4], eax
        mov     eax, DWORD PTR [ebp-4]
        mov     DWORD PTR [ebp+8], eax
        mov     esp, ebp
        pop     ebp
        ret
```

return a;

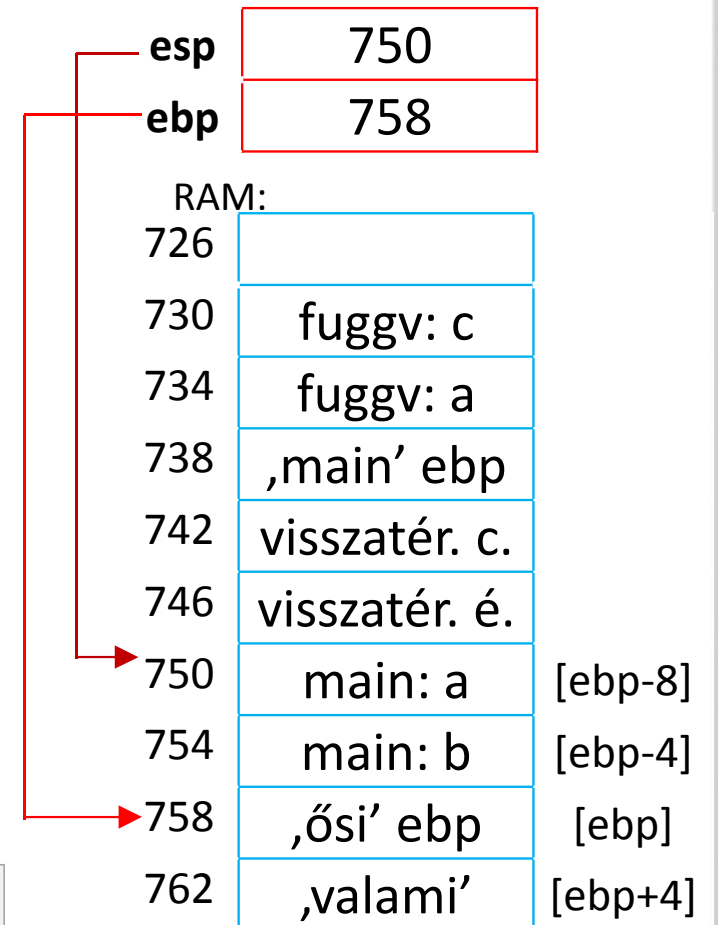


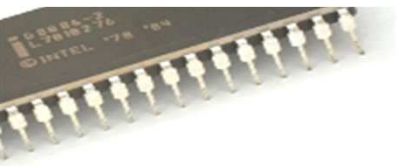


Példa assembly szinten

```
main:  push    ebp
       mov     ebp, esp
       sub     esp, 8
       mov     DWORD PTR [ebp-8], edi
       mov     eax, DWORD PTR [ebp-8]
       push   eax
       call   fuggv
       pop    eax
```

```
b=fuggv(a);
```

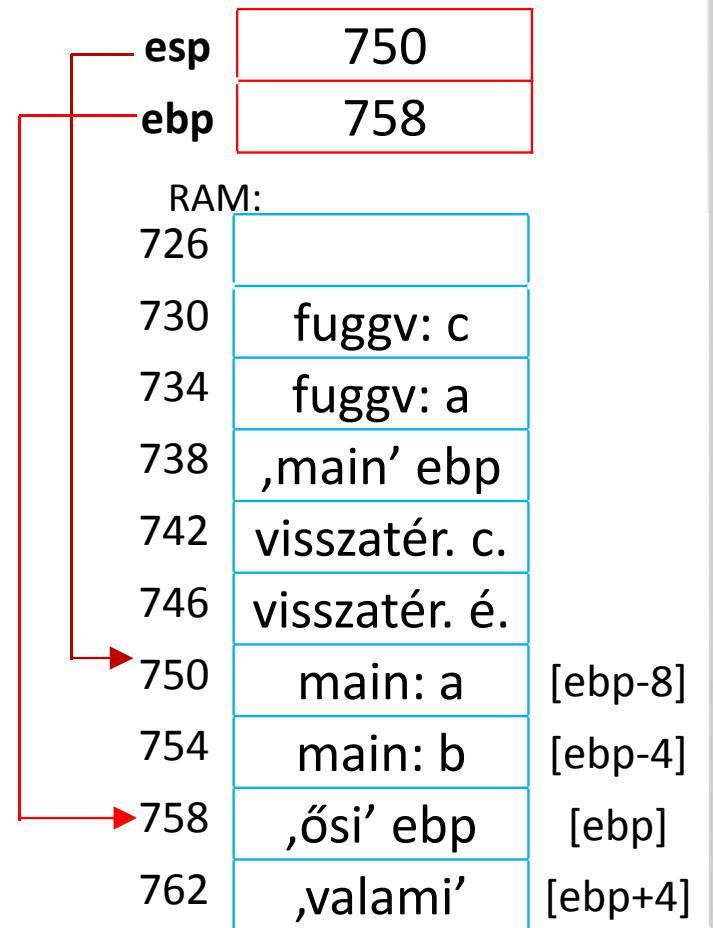


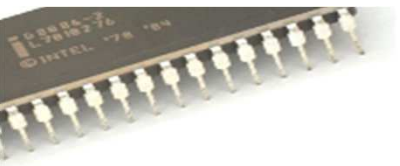


Példa assembly szinten

```
main:  push    ebp
       mov     ebp, esp
       sub     esp, 8
       mov     DWORD PTR [ebp-8], edi
       mov     eax, DWORD PTR [ebp-8]
       push   eax
       call   fuggv
       pop    eax
       mov     DWORD PTR [ebp-4], eax
```

b=fuggv(a);

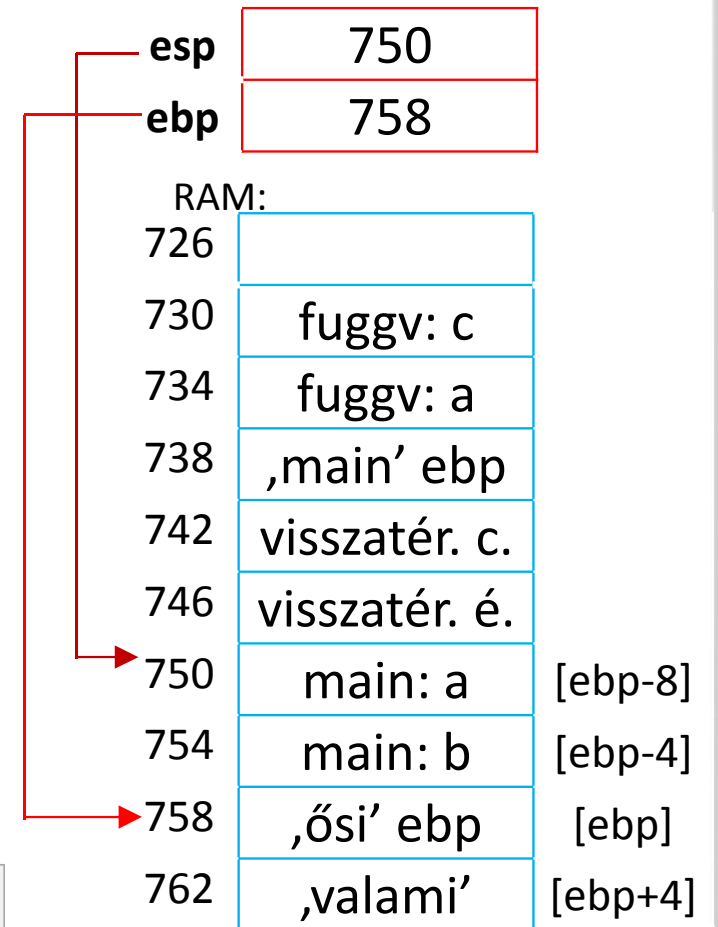


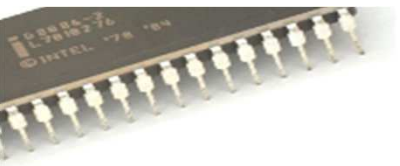


Példa assembly szinten

```
main:  push    ebp
       mov     ebp, esp
       sub     esp, 8
       mov     DWORD PTR [ebp-8], edi
       mov     eax, DWORD PTR [ebp-8]
       push   eax
       call   fuggv
       pop    eax
       mov     DWORD PTR [ebp-4], eax
       mov
```

```
return b;
```

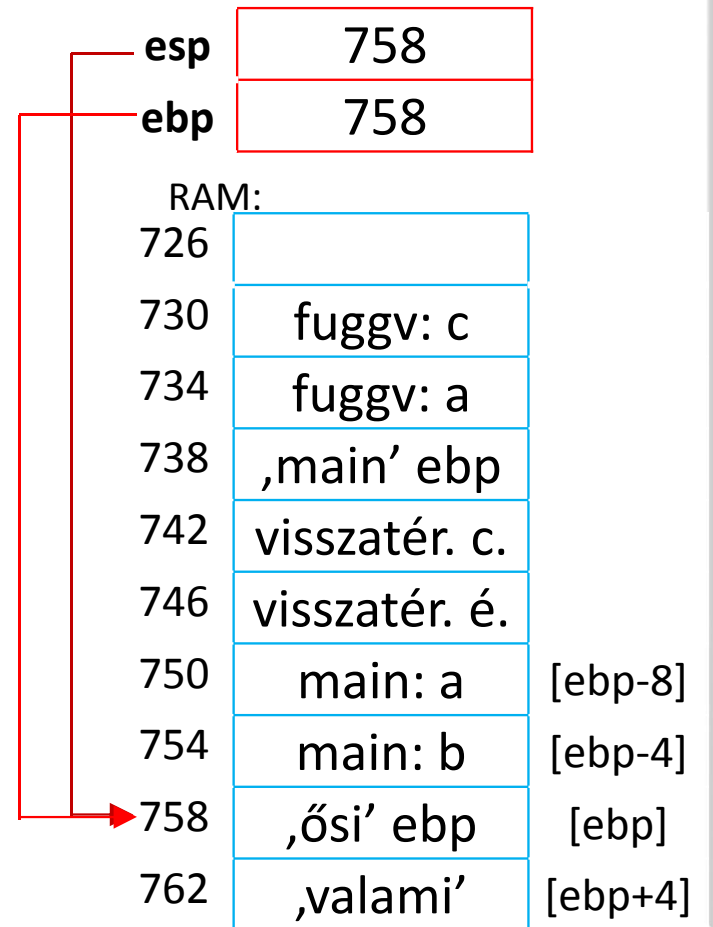


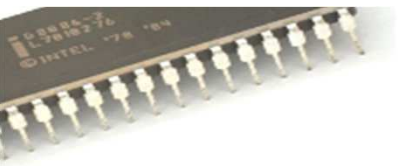


Példa assembly szinten

```
main:  push    ebp
       mov     ebp, esp
       sub     esp, 8
       mov     DWORD PTR [ebp-8], edi
       mov     eax, DWORD PTR [ebp-8]
       push   eax
       call   fuggv
       pop    eax
       mov     DWORD PTR [ebp-4], eax
       mov     eax, DWORD PTR [ebp-4]
       mov     esp, ebp
```

```
return b;
```

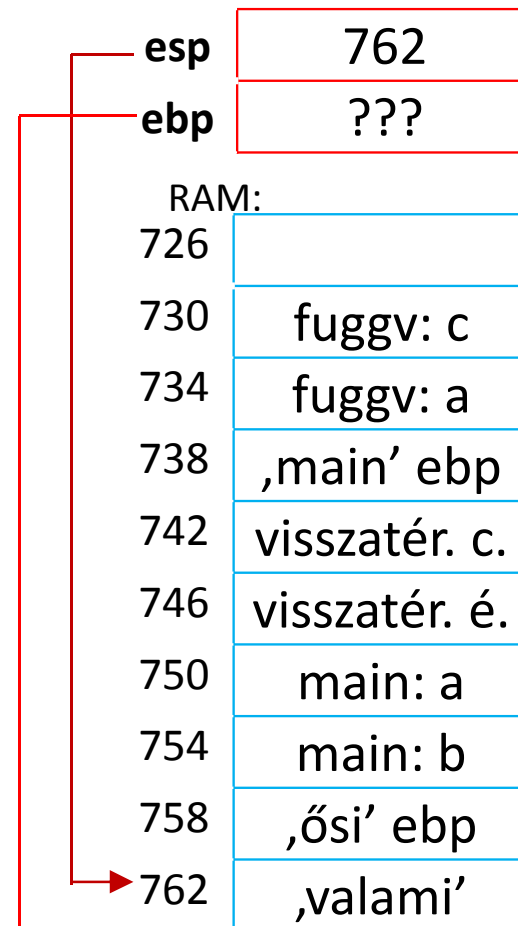


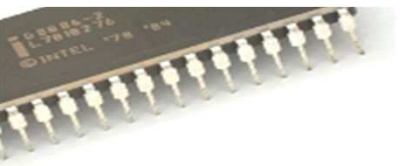


Példa assembly szinten

```
main:  push  ebp
       mov  ebp, esp
       sub  esp, 8
       mov  DWORD PTR [ebp-8], edi
       mov  eax, DWORD PTR [ebp-8]
       push eax
       call fuggv
       pop  eax
       mov  DWORD PTR [ebp-4], eax
       mov  eax, DWORD PTR [ebp-4]
       mov  esp, ebp
       pop  ebp
```

```
return b;
```





Példa assembly szinten

```
main:  push  ebp
       mov  ebp, esp
       sub  esp, 8
       mov  DWORD PTR [ebp-8], edi
       mov  eax, DWORD PTR [ebp-8]
       push eax
       call fuggv
       pop  eax
       mov  DWORD PTR [ebp-4], eax
       mov  eax, DWORD PTR [ebp-4]
       mov  esp, ebp
       pop  ebp
       ret
```

```
return b;
```

