

Лекция 5

26 февраля

```
long long f1(long long a, long long b) {  
    long long c;  
    c = a + b;  
    return c;  
}
```

```
; начало функции пропущено
```

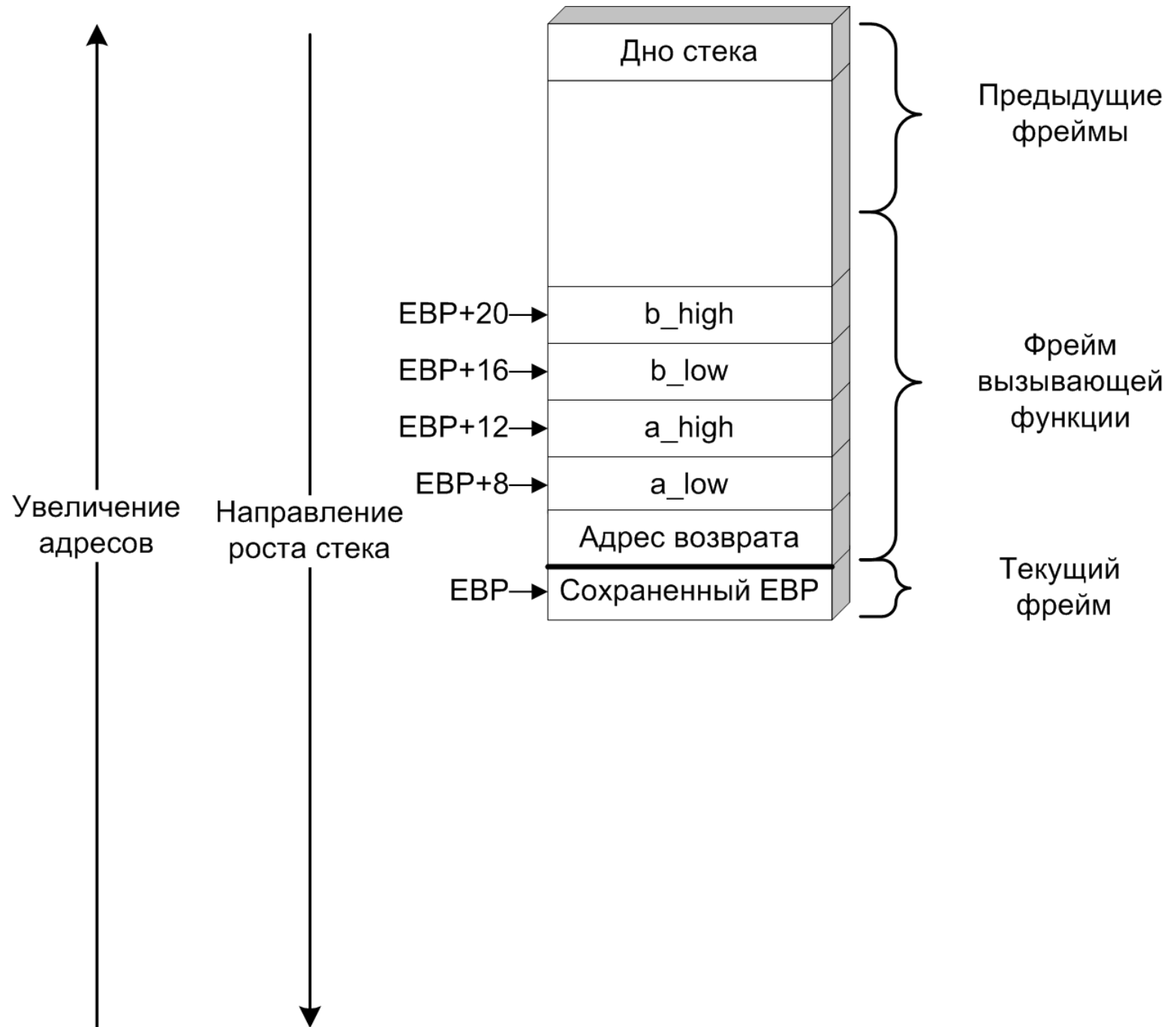
```
    mov     eax, DWORD [ebp+16] ; (1)
```

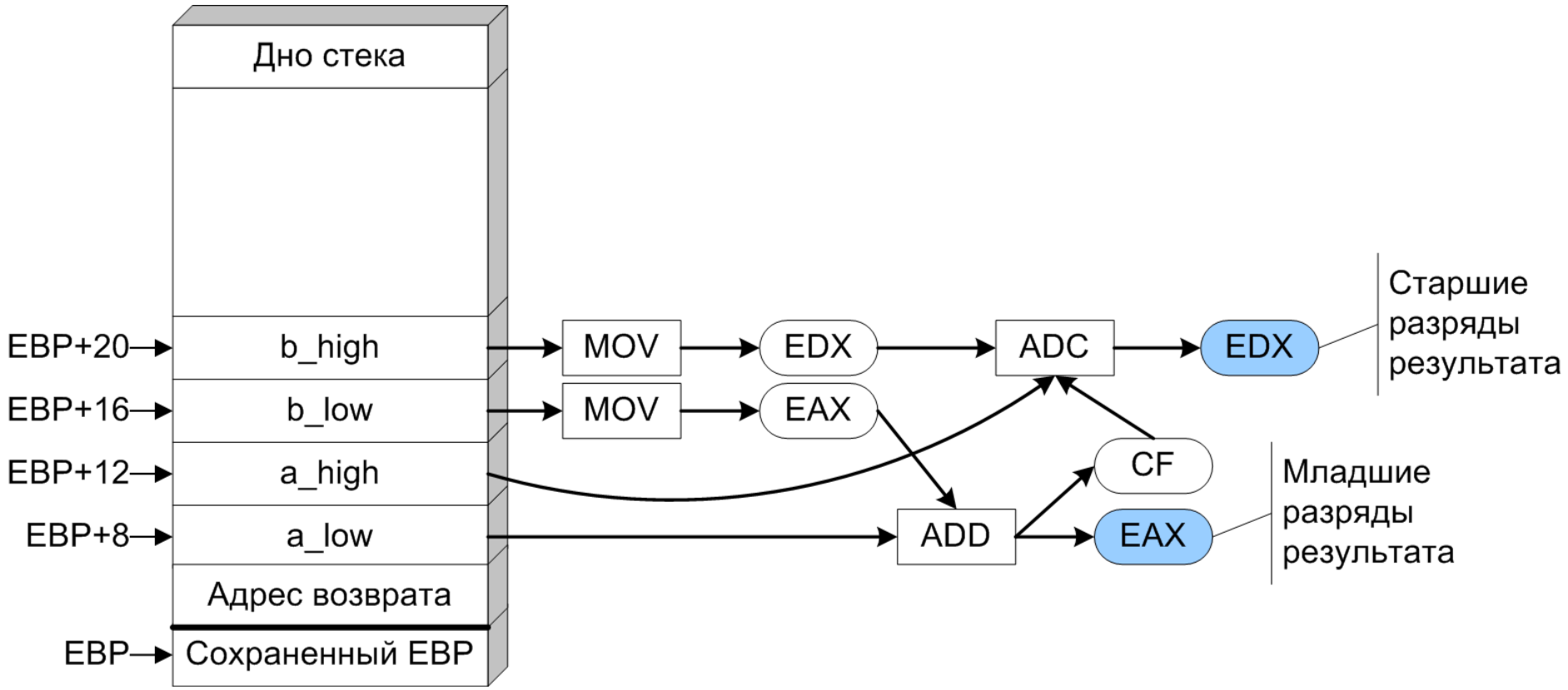
```
    mov     edx, DWORD [ebp+20] ; (2)
```

```
    add     eax, DWORD [ebp+8]  ; (3)
```

```
    adc     edx, DWORD [ebp+12] ; (4)
```

```
; конец функции пропущен
```





```
long long f3(long long a, long long b) {  
    long long c;  
    c = a - b;  
    return c;  
}
```

```
; начало функции пропущено
```

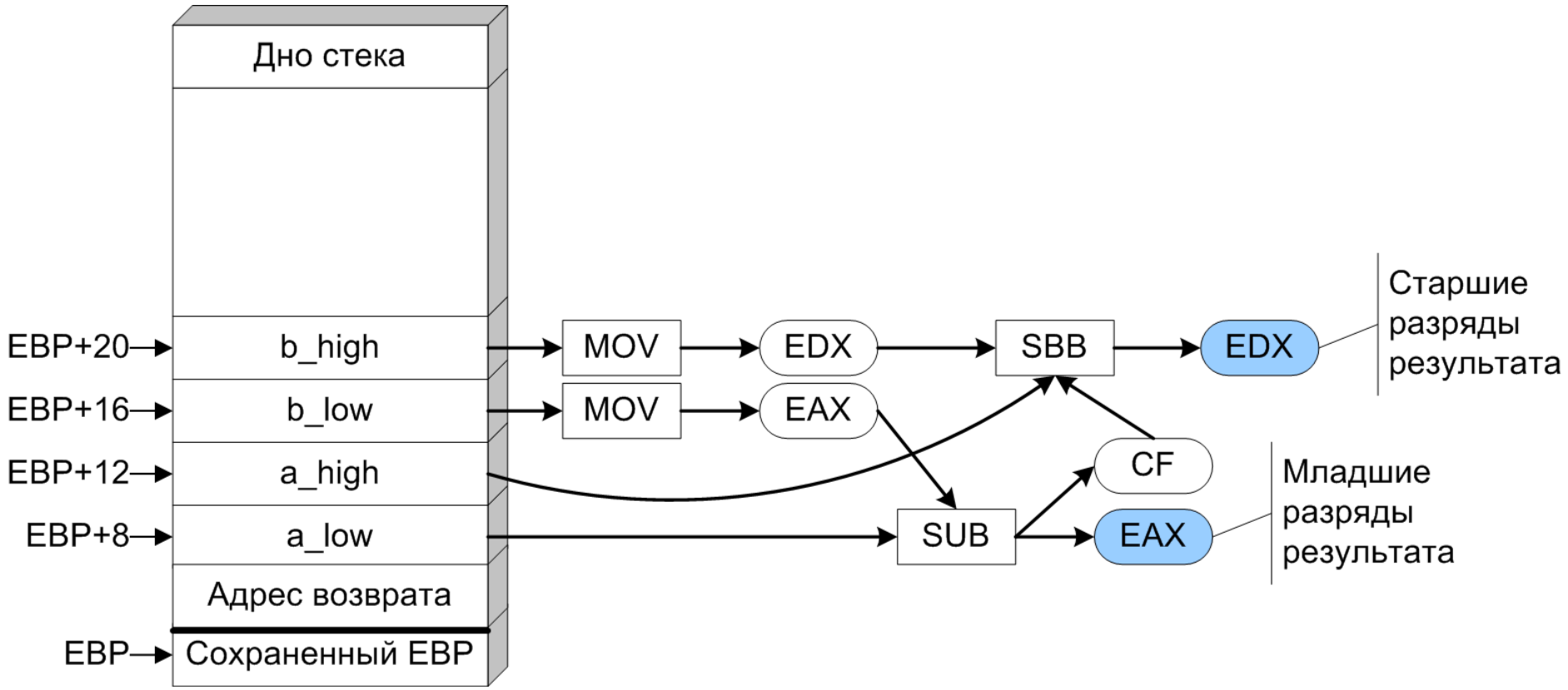
```
    mov     eax, DWORD [ebp+8] ; (1)
```

```
    mov     edx, DWORD [ebp+12] ; (2)
```

```
    sub     eax, DWORD [ebp+16] ; (3)
```

```
    sbb     edx, DWORD [ebp+20] ; (4)
```

```
; конец функции пропущен
```



```
long long f2(long long a,  
             long long b) {  
    long long c;  
    c = a * b;  
    return c;  
}
```

```
global f2  
f2:  
    push    ebp  
    mov     ebp, esp  
    sub     esp, 8  
    mov     DWORD [esp], ebx      ; (1)  
    mov     ecx, DWORD [ebp+20]  ; (2)  
    mov     ebx, DWORD [ebp+8]   ; (3)  
    mov     DWORD [esp+4], esi   ; (4)  
    mov     eax, DWORD [ebp+12]  ; (5)  
    mov     esi, DWORD [ebp+16]  ; (6)  
    imul   ecx, ebx              ; (7)  
    imul   eax, esi              ; (8)  
    add    ecx, eax              ; (9)  
    mov    eax, esi              ; (10)  
    mul   ebx                    ; (11)  
    mov   ebx, DWORD [esp]       ; (12)  
    lea  esi, [ecx+edx]          ; (13)  
    mov  edx, esi                ; (14)  
    mov  esi, DWORD [esp+4]     ; (15)  
    mov  esp, ebp  
    pop  ebp  
    ret
```

