

```

; Attributes: bp-based frame
sub_405D3E proc near

var_1C= word ptr -1Ch
var_19= byte ptr -19h
var_4= dword ptr -4
arg_0= dword ptr 8
arg_4= dword ptr 0Ch
arg_8= byte ptr 10h

push    ebp
mov     ebp, esp
sub    esp, 1Ch
push    edi
push    esi
push    ebx
mov     [ebp+var_4], eax
mov     eax, 13h
cmp     cl, 0
jnz    short loc_405D6B

```

```

    mov     eax, [ebp+arg_4]
    cmp     eax, 2
    jge    short loc_405D61

```

```

        mov     eax, 2

```

```

loc_405D61:
    cmp     eax, 12h
    jle    short loc_405D6B

```

```

        mov     eax, 12h

```

```

loc_405D6B:
    mov     [ebp+arg_4], eax
    push    eax
    mov     eax, 270Fh
    cmp     [ebp+arg_8], 2
    jb     short loc_405D7D

```

```

    mov     eax, [ebp+arg_0]

```

```

loc_405D7D:
    push    eax
    lea     eax, [ebp+var_1C]
    call    sub_4062DA
    mov     edi, [ebp+var_4]
    movzx   eax, [ebp+var_1C]
    sub     eax, 7FFFh
    cmp     eax, 2
    jnb    short loc_405DA7

```

```

loc_405DA7:
    lea     esi, [ebp+var_19]
    movzx   ebx, [ebp+arg_8]
    cmp     bl, 1
    jz     short loc_405DC3

```

```

    cmp     bl, 4
    ja     short loc_405DC1

```

```

    movsx   eax, [ebp+var_1C]
    cmp     eax, [ebp+arg_4]
    jie    short loc_405DC3

```

```

loc_405DC1:
    mov     bl, 0

```

```

lea     esi, loc_405D38[eax+eax*2]
mov     ecx, 3
rep    movsb
jmp    short loc_405DCA

```

```

loc_405DC3:
    call    off_405DD8[ebx*4]

```

```

loc_405DCA:
    mov     eax, edi
    sub    eax, [ebp+var_4]
    pop    ebx
    pop    esi
    pop    edi
    mov     esp, ebp
    pop    ebp
    ret    0Ch
    sub_405D3E endp

```