|  |  |  |
| --- | --- | --- |
| **Source (C/C++)** | **Assembler (assembly language)** | **Machine code** |
| int myArray[] = { 1, 2, 3, 4, 5 }; | mov DWORD PTR myArray$[rsp], 1  mov DWORD PTR myArray$[rsp+4], 2  mov DWORD PTR myArray$[rsp+8], 3  mov DWORD PTR myArray$[rsp+12], 4  mov DWORD PTR myArray$[rsp+16], 5 | c7 44 24 28 01  c7 44 24 2c 02  c7 44 24 30 03  c7 44 24 34 04  c7 44 24 38 05 |
| int total = 0; | mov DWORD PTR total$[rsp], 0 | c7 44 24 24 00 |
| for ( unsigned int index = 0; index != 5; index++ ) | mov DWORD PTR index$1[rsp], 0  jmp SHORT $LN4@main  mov eax, DWORD PTR index$1[rsp]  inc eax  mov DWORD PTR index$1[rsp], eax  cmp DWORD PTR index$1[rsp], 5  je SHORT $LN3@main | c7 44 24 20 00  eb 0a  8b 44 24 20  ff c0  89 44 24 20  83 7c 24 20 05  74 16 |
| {  total += myArray[index]; | mov eax, DWORD PTR index$1[rsp]  mov eax, DWORD PTR myArray$[rsp+rax\*4]  mov ecx, DWORD PTR total$[rsp]  add ecx, eax  mov eax, ecx  mov DWORD PTR total$[rsp], eax | 8b 44 24 20  8b 44 84 28  8b 4c 24 24  03 c8  8b c1  89 44 24 24 |
| } | jmp SHORT | eb d9 |