# **CS 410 Binary to C++ With Security Vulnerabilities Activity Template**

Janera Dobson

Southern New Hampshire University

CS-410-T3249

**Step 1:** Convert the binary file to assembly code.

**(gdb) disassemble DisplayMenu**

**Dump of assembler code for function \_Z11DisplayMenuv:**

**0x0000000000000000 <+0>: push %rbp**

**0x0000000000000001 <+1>: mov %rsp,%rbp**

**0x0000000000000004 <+4>: lea 0x0(%rip),%rsi # 0xb <\_Z11DisplayMenuv+11>**

**0x000000000000000b <+11>: lea 0x0(%rip),%rdi # 0x12 <\_Z11DisplayMenuv+18>**

**0x0000000000000012 <+18>: callq 0x17 <\_Z11DisplayMenuv+23>**

**0x0000000000000017 <+23>: lea 0x0(%rip),%rsi # 0x1e <\_Z11DisplayMenuv+30>**

**0x000000000000001e <+30>: lea 0x0(%rip),%rdi # 0x25 <\_Z11DisplayMenuv+37>**

**0x0000000000000025 <+37>: callq 0x2a <\_Z11DisplayMenuv+42>**

**0x000000000000002a <+42>: lea 0x0(%rip),%rsi # 0x31 <\_Z11DisplayMenuv+49>**

**0x0000000000000031 <+49>: lea 0x0(%rip),%rdi # 0x38 <\_Z11DisplayMenuv+56>**

**0x0000000000000038 <+56>: callq 0x3d <\_Z11DisplayMenuv+61>**

**0x000000000000003d <+61>: lea 0x0(%rip),%rsi # 0x44 <\_Z11DisplayMenuv+68>**

**0x0000000000000044 <+68>: lea 0x0(%rip),%rdi # 0x4b <\_Z11DisplayMenuv+75>**

**0x000000000000004b <+75>: callq 0x50 <\_Z11DisplayMenuv+80>**

**0x0000000000000050 <+80>: lea 0x0(%rip),%rsi # 0x57 <\_Z11DisplayMenuv+87>**

**0x0000000000000057 <+87>: lea 0x0(%rip),%rdi # 0x5e <\_Z11DisplayMenuv+94>**

**0x000000000000005e <+94>: callq 0x63 <\_Z11DisplayMenuv+99>**

**0x0000000000000063 <+99>: lea 0x0(%rip),%rsi # 0x6a <\_Z11DisplayMenuv+106>**

**0x000000000000006a <+106>: lea 0x0(%rip),%rdi # 0x71 <\_Z11DisplayMenuv+113>**

**0x0000000000000071 <+113>: callq 0x76 <\_Z11DisplayMenuv+118>**

**0x0000000000000076 <+118>: nop**

**0x0000000000000077 <+119>: pop %rbp**

**0x0000000000000078 <+120>: retq**

**End of assembler dump.**

**(gdb) disassemble main**

**Dump of assembler code for function main:**

**0x0000000000000079 <+0>: push %rbp**

**0x000000000000007a <+1>: mov %rsp,%rbp**

**0x000000000000007d <+4>: sub $0x20,%rsp**

**0x0000000000000081 <+8>: mov %fs:0x28,%rax**

**0x000000000000008a <+17>: mov %rax,-0x8(%rbp)**

**0x000000000000008e <+21>: xor %eax,%eax**

**0x0000000000000090 <+23>: movl $0x0,-0x14(%rbp)**

**0x0000000000000097 <+30>: mov -0x14(%rbp),%eax**

**0x000000000000009a <+33>: cmp $0x5,%eax**

**0x000000000000009d <+36>: je 0x308 <main+655>**

**0x00000000000000a3 <+42>: lea 0x0(%rip),%rsi # 0xaa <main+49>**

**0x00000000000000aa <+49>: lea 0x0(%rip),%rdi # 0xb1 <main+56>**

**0x00000000000000b1 <+56>: callq 0xb6 <main+61>**

**0x00000000000000b6 <+61>: lea 0x0(%rip),%rsi # 0xbd <main+68>**

**0x00000000000000bd <+68>: lea 0x0(%rip),%rdi # 0xc4 <main+75>**

**0x00000000000000c4 <+75>: callq 0xc9 <main+80>**

**0x00000000000000c9 <+80>: lea 0x0(%rip),%rsi # 0xd0 <main+87>**

**0x00000000000000d0 <+87>: lea 0x0(%rip),%rdi # 0xd7 <main+94>**

**0x00000000000000d7 <+94>: callq 0xdc <main+99>**

**0x00000000000000dc <+99>: lea 0x0(%rip),%rsi # 0xe3 <main+106>**

**0x00000000000000e3 <+106>: lea 0x0(%rip),%rdi # 0xea <main+113>**

**0x00000000000000ea <+113>: callq 0xef <main+118>**

**0x00000000000000ef <+118>: lea 0x0(%rip),%rsi # 0xf6 <main+125>**

**0x00000000000000f6 <+125>: lea 0x0(%rip),%rdi # 0xfd <main+132>**

**0x00000000000000fd <+132>: callq 0x102 <main+137>**

**0x0000000000000102 <+137>: lea 0x0(%rip),%rsi # 0x109 <main+144>**

**0x0000000000000109 <+144>: lea 0x0(%rip),%rdi # 0x110 <main+151>**

**0x0000000000000110 <+151>: callq 0x115 <main+156>**

**0x0000000000000115 <+156>: lea -0x14(%rbp),%rax**

**0x0000000000000119 <+160>: mov %rax,%rsi**

**0x000000000000011c <+163>: lea 0x0(%rip),%rdi # 0x123 <main+170>**

**0x0000000000000123 <+170>: callq 0x128 <main+175>**

**0x0000000000000128 <+175>: mov -0x14(%rbp),%eax**

**0x000000000000012b <+178>: cmp $0x1,%eax**

**0x000000000000012e <+181>: jne 0x1c9 <main+336>**

**0x0000000000000134 <+187>: lea -0x10(%rbp),%rax**

**0x0000000000000138 <+191>: mov %rax,%rsi**

**0x000000000000013b <+194>: lea 0x0(%rip),%rdi # 0x142 <main+201>**

**0x0000000000000142 <+201>: callq 0x147 <main+206>**

**0x0000000000000147 <+206>: mov %rax,%rdx**

**0x000000000000014a <+209>: lea -0xc(%rbp),%rax**

**0x000000000000014e <+213>: mov %rax,%rsi**

**0x0000000000000151 <+216>: mov %rdx,%rdi**

**0x0000000000000154 <+219>: callq 0x159 <main+224>**

**0x0000000000000159 <+224>: mov -0x10(%rbp),%eax**

**0x000000000000015c <+227>: mov %eax,%esi**

**0x000000000000015e <+229>: lea 0x0(%rip),%rdi # 0x165 <main+236>**

**0x0000000000000165 <+236>: callq 0x16a <main+241>**

**---Type <return> to continue, or q <return> to quit---return**

**0x000000000000016a <+241>: lea 0x0(%rip),%rsi # 0x171 <main+248>**

**0x0000000000000171 <+248>: mov %rax,%rdi**

**0x0000000000000174 <+251>: callq 0x179 <main+256>**

**0x0000000000000179 <+256>: mov %rax,%rdx**

**0x000000000000017c <+259>: mov -0xc(%rbp),%eax**

**0x000000000000017f <+262>: mov %eax,%esi**

**0x0000000000000181 <+264>: mov %rdx,%rdi**

**0x0000000000000184 <+267>: callq 0x189 <main+272>**

**0x0000000000000189 <+272>: lea 0x0(%rip),%rsi # 0x190 <main+279>**

**0x0000000000000190 <+279>: mov %rax,%rdi**

**0x0000000000000193 <+282>: callq 0x198 <main+287>**

**0x0000000000000198 <+287>: mov %rax,%rcx**

**0x000000000000019b <+290>: mov -0x10(%rbp),%edx**

**0x000000000000019e <+293>: mov -0xc(%rbp),%eax**

**0x00000000000001a1 <+296>: sub %eax,%edx**

**0x00000000000001a3 <+298>: mov %edx,%eax**

**0x00000000000001a5 <+300>: mov %eax,%esi**

**0x00000000000001a7 <+302>: mov %rcx,%rdi**

**0x00000000000001aa <+305>: callq 0x1af <main+310>**

**0x00000000000001af <+310>: mov %rax,%rdx**

**0x00000000000001b2 <+313>: mov 0x0(%rip),%rax # 0x1b9 <main+320>**

**0x00000000000001b9 <+320>: mov %rax,%rsi**

**0x00000000000001bc <+323>: mov %rdx,%rdi**

**0x00000000000001bf <+326>: callq 0x1c4 <main+331>**

**0x00000000000001c4 <+331>: jmpq 0x97 <main+30>**

**0x00000000000001c9 <+336>: mov -0x14(%rbp),%eax**

**0x00000000000001cc <+339>: cmp $0x2,%eax**

**0x00000000000001cf <+342>: jne 0x268 <main+495>**

**0x00000000000001d5 <+348>: lea -0x10(%rbp),%rax**

**0x00000000000001d9 <+352>: mov %rax,%rsi**

**0x00000000000001dc <+355>: lea 0x0(%rip),%rdi # 0x1e3 <main+362>**

**0x00000000000001e3 <+362>: callq 0x1e8 <main+367>**

**0x00000000000001e8 <+367>: mov %rax,%rdx**

**0x00000000000001eb <+370>: lea -0xc(%rbp),%rax**

**0x00000000000001ef <+374>: mov %rax,%rsi**

**0x00000000000001f2 <+377>: mov %rdx,%rdi**

**0x00000000000001f5 <+380>: callq 0x1fa <main+385>**

**0x00000000000001fa <+385>: mov -0x10(%rbp),%eax**

**0x00000000000001fd <+388>: mov %eax,%esi**

**0x00000000000001ff <+390>: lea 0x0(%rip),%rdi # 0x206 <main+397>**

**0x0000000000000206 <+397>: callq 0x20b <main+402>**

**0x000000000000020b <+402>: lea 0x0(%rip),%rsi # 0x212 <main+409>**

**0x0000000000000212 <+409>: mov %rax,%rdi**

**0x0000000000000215 <+412>: callq 0x21a <main+417>**

**0x000000000000021a <+417>: mov %rax,%rdx**

**0x000000000000021d <+420>: mov -0xc(%rbp),%eax**

**0x0000000000000220 <+423>: mov %eax,%esi**

**0x0000000000000222 <+425>: mov %rdx,%rdi**

**0x0000000000000225 <+428>: callq 0x22a <main+433>**

**---Type <return> to continue, or q <return> to quit---return**

**0x000000000000022a <+433>: lea 0x0(%rip),%rsi # 0x231 <main+440>**

**0x0000000000000231 <+440>: mov %rax,%rdi**

**0x0000000000000234 <+443>: callq 0x239 <main+448>**

**0x0000000000000239 <+448>: mov %rax,%rcx**

**0x000000000000023c <+451>: mov -0x10(%rbp),%edx**

**0x000000000000023f <+454>: mov -0xc(%rbp),%eax**

**0x0000000000000242 <+457>: add %edx,%eax**

**0x0000000000000244 <+459>: mov %eax,%esi**

**0x0000000000000246 <+461>: mov %rcx,%rdi**

**0x0000000000000249 <+464>: callq 0x24e <main+469>**

**0x000000000000024e <+469>: mov %rax,%rdx**

**0x0000000000000251 <+472>: mov 0x0(%rip),%rax # 0x258 <main+479>**

**0x0000000000000258 <+479>: mov %rax,%rsi**

**0x000000000000025b <+482>: mov %rdx,%rdi**

**0x000000000000025e <+485>: callq 0x263 <main+490>**

**0x0000000000000263 <+490>: jmpq 0x97 <main+30>**

**0x0000000000000268 <+495>: mov -0x14(%rbp),%eax**

**0x000000000000026b <+498>: cmp $0x3,%eax**

**0x000000000000026e <+501>: jne 0x97 <main+30>**

**0x0000000000000274 <+507>: lea -0x10(%rbp),%rax**

**0x0000000000000278 <+511>: mov %rax,%rsi**

**0x000000000000027b <+514>: lea 0x0(%rip),%rdi # 0x282 <main+521>**

**0x0000000000000282 <+521>: callq 0x287 <main+526>**

**0x0000000000000287 <+526>: mov %rax,%rdx**

**0x000000000000028a <+529>: lea -0xc(%rbp),%rax**

**0x000000000000028e <+533>: mov %rax,%rsi**

**0x0000000000000291 <+536>: mov %rdx,%rdi**

**0x0000000000000294 <+539>: callq 0x299 <main+544>**

**0x0000000000000299 <+544>: mov -0x10(%rbp),%eax**

**0x000000000000029c <+547>: mov %eax,%esi**

**0x000000000000029e <+549>: lea 0x0(%rip),%rdi # 0x2a5 <main+556>**

**0x00000000000002a5 <+556>: callq 0x2aa <main+561>**

**0x00000000000002aa <+561>: lea 0x0(%rip),%rsi # 0x2b1 <main+568>**

**0x00000000000002b1 <+568>: mov %rax,%rdi**

**0x00000000000002b4 <+571>: callq 0x2b9 <main+576>**

**0x00000000000002b9 <+576>: mov %rax,%rdx**

**0x00000000000002bc <+579>: mov -0xc(%rbp),%eax**

**0x00000000000002bf <+582>: mov %eax,%esi**

**0x00000000000002c1 <+584>: mov %rdx,%rdi**

**0x00000000000002c4 <+587>: callq 0x2c9 <main+592>**

**0x00000000000002c9 <+592>: lea 0x0(%rip),%rsi # 0x2d0 <main+599>**

**0x00000000000002d0 <+599>: mov %rax,%rdi**

**0x00000000000002d3 <+602>: callq 0x2d8 <main+607>**

**0x00000000000002d8 <+607>: mov %rax,%rcx**

**0x00000000000002db <+610>: mov -0x10(%rbp),%eax**

**0x00000000000002de <+613>: mov -0xc(%rbp),%esi**

**0x00000000000002e1 <+616>: cltd**

**0x00000000000002e2 <+617>: idiv %esi**

**0x00000000000002e4 <+619>: mov %eax,%esi**

**---Type <return> to continue, or q <return> to quit---return**

**0x00000000000002e6 <+621>: mov %rcx,%rdi**

**0x00000000000002e9 <+624>: callq 0x2ee <main+629>**

**0x00000000000002ee <+629>: mov %rax,%rdx**

**0x00000000000002f1 <+632>: mov 0x0(%rip),%rax # 0x2f8 <main+639>**

**0x00000000000002f8 <+639>: mov %rax,%rsi**

**0x00000000000002fb <+642>: mov %rdx,%rdi**

**0x00000000000002fe <+645>: callq 0x303 <main+650>**

**0x0000000000000303 <+650>: jmpq 0x97 <main+30>**

**0x0000000000000308 <+655>: mov $0x0,%eax**

**0x000000000000030d <+660>: mov -0x8(%rbp),%rcx**

**0x0000000000000311 <+664>: xor %fs:0x28,%rcx**

**0x000000000000031a <+673>: je 0x321 <main+680>**

**0x000000000000031c <+675>: callq 0x321 <main+680>**

**0x0000000000000321 <+680>: leaveq**

**0x0000000000000322 <+681>: retq**

**End of assembler dump.**

**(gdb)**

**Step 2:** Explain the functionality of the blocks of assembly code.

| **Blocks of Assembly Code** | **Explanation of Functionality** |
| --- | --- |
| **push %rbp**  **mov %rsp,%rbp**  **lea 0x0(%rip),%rsi # 0xb <\_Z11DisplayMenuv+11>**  **lea 0x0(%rip),%rdi # 0x12 <\_Z11DisplayMenuv+18>**  **callq 0x17 <\_Z11DisplayMenuv+23>**  **lea 0x0(%rip),%rsi # 0x1e <\_Z11DisplayMenuv+30>**  **lea 0x0(%rip),%rdi # 0x25 <\_Z11DisplayMenuv+37>**  **callq 0x2a <\_Z11DisplayMenuv+42>**  **lea 0x0(%rip),%rsi # 0x31 <\_Z11DisplayMenuv+49>**  **lea 0x0(%rip),%rdi # 0x38 <\_Z11DisplayMenuv+56>**  **callq 0x3d <\_Z11DisplayMenuv+61>**  **lea 0x0(%rip),%rsi # 0x44 <\_Z11DisplayMenuv+68>**  **lea 0x0(%rip),%rdi # 0x4b <\_Z11DisplayMenuv+75>**  **callq 0x50 <\_Z11DisplayMenuv+80>**  **lea 0x0(%rip),%rsi # 0x57 <\_Z11DisplayMenuv+87>**  **lea 0x0(%rip),%rdi # 0x5e <\_Z11DisplayMenuv+94>**  **callq 0x63 <\_Z11DisplayMenuv+99>**  **lea 0x0(%rip),%rsi # 0x6a <\_Z11DisplayMenuv+106>**  **lea 0x0(%rip),%rdi # 0x71 <\_Z11DisplayMenuv+113>**  **callq 0x76 <\_Z11DisplayMenuv+118>**  **pop %rbp**  **retq** | Push %rbp to current stack  Move %rbp in %rsp  Load %rsi in 0(%rip)  Load %rdi in 0(%rip)  Call DIsplayMenu()  Load %rsi in 0(%rip)  Load %rdi in 0(%rip)  Call DIsplayMenu()  Load %rsi in 0(%rip)  Load %rdi in 0(%rip)  Call DisplayMenu()  Load %rsi in 0(%rip)  Load %rdi in 0(%rip)  Call DisplayMenu()  Load %rsi in 0(%rip)  Load %rdi in 0(%rip)  Call DisplayMenu()  Load %rsi in 0(%rip)  Load %rdi in 0(%rip)  Call DisplayMenu()  Removes 4-byte data in %rbp  Return |
| **mov -0x14(%rbp),%eax**  **cmp $0x5,%eax**  **je 0x308 <main+655>**  **lea 0x0(%rip),%rsi # 0xaa <main+49>**  **lea 0x0(%rip),%rdi # 0xb1 <main+56>**  **callq 0xb6 <main+61>** | Move %eax in -14(%rbp)  Compare %eax to 5  Jump if equal <main+655>  Load %rsi in 0(%rip)  Load %rdi in 0(rip)  Call <main+61> |
| **lea -0x14(%rbp),%rax**  **mov %rax,%rsi**  **lea 0x0(%rip),%rdi # 0x123 <main+170>**  **callq 0x128 <main+175>** | Load %rax in 14(%rbp)  Move %rsi in %rax  Load %rdi in 0(%rip)  Call <main+175> |
| **mov -0x14(%rbp),%eax**  **cmp $0x1,%eax**  **jne 0x1c9 <main+336>**  **lea -0x10(%rbp),%rax**  **mov %rax,%rsi**  **lea 0x0(%rip),%rdi # 0x142 <main+201>**  **callq 0x147 <main+206>**  **mov %rax,%rcx**  **mov -0x10(%rbp),%edx**  **mov -0xc(%rbp),%eax**  **sub %eax,%edx**  **mov %edx,%eax**  **mov %eax,%esi**  **mov %rcx,%rdi**  **callq 0x1af <main+310>**  **mov -0x14(%rbp),%eax**  **cmp $0x2,%eax**  **jne 0x268 <main+495>**  **lea -0x10(%rbp),%rax**  **mov %rax,%rsi**  **lea 0x0(%rip),%rdi # 0x1e3 <main+362>**  **callq 0x1e8 <main+367>**  **mov %rax,%rcx**  **mov -0x10(%rbp),%edx**  **mov -0xc(%rbp),%eax**  **add %edx,%eax**  **mov %eax,%esi**  **mov %rcx,%rdi**  **callq 0x24e <main+469>**  **mov -0x14(%rbp),%eax**  **cmp $0x3,%eax**  **jne 0x97 <main+30>**  **lea -0x10(%rbp),%rax**  **mov %rax,%rsi**  **lea 0x0(%rip),%rdi # 0x282 <main+521>**  **callq 0x287 <main+526>**  **mov %rax,%rcx**  **mov -0x10(%rbp),%eax**  **mov -0xc(%rbp),%esi**  **cltd**  **idiv %esi**  **mov %eax,%esi**  **mov %rcx,%rdi**  **callq 0x2ee <main+629>** | Move %eax in -14(%rbp)  Compare %eax to 1  Conditional jump following previous test to <main+336>  Load %rax in -10(%rbp)  Move %rsi in %rax  Load %rdi in 0(%rip)  Call <main+206>  Move %rcx in %rax  Move %edx in -10(%rbp)  Move %eax in %rbp  Subtract %edx in %eax  Move %eax in %edx  Move %esi in %eax  Move %rdi in %rcx  Call <main+310>  Move %eax in -14(%rbp)  Compare %eax to 2  Conditional jump following previous test to <main+495>  Load %rax in -10(%rbp)  Move %rsi in %rax  Load %rdi in 0(%rip)  Call <main+367>  Move %rcx in %rax  Move %edx in -10(%rbp)  Move %eax in %rbp  Add %edx in %eax  Move %eax in %edx  Move %rdi in %rcx  Call <main+469>  Move %eax in -14(%rbp)  Compare %eax to 3  Conditional jump following previous test to <main+30>  Load %rax in -10(%rbp)  Move %rsi in %rax  Load %rdi in 0(%rip)  Call <main+526>  Move %rcx in %rax  Move %eax in -10(%rbp)  Move %esi in %rbp  Convert signed long to sing double long  Signed divide %esi  Move %esi in %eax  Move %rdi in %rcx  Call <main+629> |
| **mov %rax,%rsi**  **mov %rdx,%rdi**  **callq 0x303 <main+650>**  **jmpq 0x97 <main+30>**  **mov $0x0,%eax**  **mov -0x8(%rbp),%rcx**  **xor %fs:0x28,%rcx**  **je 0x321 <main+680>**  **callq 0x321 <main+680>**  **leaveq**  **retq** | Move %rsi in %rax  Move %rdi in %rdx  Call <main+650>  Unconditional jump <main+30>  Move %eax in 0  Move %rcx in -8(%rbp)  Bitwise operation in %rcx  Jump if equal <main+680>  Call <main+680>  Leave / Enter  Return |

**Step 3:** Convert the assembly code to binary.

7F 45 4C 46 02 01 01 00 00 00 00 00 00 00 00 00 01 00 3E 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 60 12 00 00 00 00 00 00 00 00 00 00 40 00 00 00 00 00 40 00 0F 00 0E 00 55 48 89 E5 48 8D 35 00 00 00 00 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 35 00 00 00 00 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 35 00 00 00 00 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 35 00 00 00 00 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 35 00 00 00 00 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 35 00 00 00 00 48 8D 3D 00 00 00 00 E8 00 00 00 00 90 5D C3 55 48 89 E5 48 83 EC 20 64 48 8B 04 25 28 00 00 00 48 89 45 F8 31 C0 C7 45 EC 00 00 00 00 8B 45 EC 83 F8 05 0F 84 65 02 00 00 48 8D 35 00 00 00 00 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 35 00 00 00 00 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 35 00 00 00 00 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 35 00 00 00 00 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 35 00 00 00 00 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 35 00 00 00 00 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 45 EC 48 89 C6 48 8D 3D 00 00 00 00 E8 00 00 00 00 8B 45 EC 83 F8 01 0F 85 95 00 00 00 48 8D 45 F0 48 89 C6 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 89 C2 48 8D 45 F4 48 89 C6 48 89 D7 E8 00 00 00 00 8B 45 F0 89 C6 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 35 00 00 00 00 48 89 C7 E8 00 00 00 00 48 89 C2 8B 45 F4 89 C6 48 89 D7 E8 00 00 00 00 48 8D 35 00 00 00 00 48 89 C7 E8 00 00 00 00 48 89 C1 8B 55 F0 8B 45 F4 29 C2 89 D0 89 C6 48 89 CF E8 00 00 00 00 48 89 C2 48 8B 05 00 00 00 00 48 89 C6 48 89 D7 E8 00 00 00 00 E9 CE FE FF FF 8B 45 EC 83 F8 02 0F 85 93 00 00 00 48 8D 45 F0 48 89 C6 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 89 C2 48 8D 45 F4 48 89 C6 48 89 D7 E8 00 00 00 00 8B 45 F0 89 C6 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 35 00 00 00 00 48 89 C7 E8 00 00 00 00 48 89 C2 8B 45 F4 89 C6 48 89 D7 E8 00 00 00 00 48 8D 35 00 00 00 00 48 89 C7 E8 00 00 00 00 48 89 C1 8B 55 F0 8B 45 F4 01 D0 89 C6 48 89 CF E8 00 00 00 00 48 89 C2 48 8B 05 00 00 00 00 48 89 C6 48 89 D7 E8 00 00 00 00 E9 2F FE FF FF 8B 45 EC 83 F8 03 0F 85 23 FE FF FF 48 8D 45 F0 48 89 C6 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 89 C2 48 8D 45 F4 48 89 C6 48 89 D7 E8 00 00 00 00 8B 45 F0 89 C6 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 35 00 00 00 00 48 89 C7 E8 00 00 00 00 48 89 C2 8B 45 F4 89 C6 48 89 D7 E8 00 00 00 00 48 8D 35 00 00 00 00 48 89 C7 E8 00 00 00 00 48 89 C1 8B 45 F0 8B 75 F4 99 F7 FE 89 C6 48 89 CF E8 00 00 00 00 48 89 C2 48 8B 05 00 00 00 00 48 89 C6 48 89 D7 E8 00 00 00 00 E9 8F FD FF FF B8 00 00 00 00 48 8B 4D F8 64 48 33 0C 25 28 00 00 00 74 05 E8 00 00 00 00 C9 C3 55 48 89 E5 48 83 EC 10 89 7D FC 89 75 F8 83 7D FC 01 75 32 81 7D F8 FF FF 00 00 75 29 48 8D 3D 00 00 00 00 E8 00 00 00 00 48 8D 15 00 00 00 00 48 8D 35 00 00 00 00 48 8B 05 00 00 00 00 48 89 C7 E8 00 00 00 00 90 C9 C3 55 48 89 E5 BE FF FF 00 00 BF 01 00 00 00 E8 A4 FF FF FF 5D C3 00 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 00 2D 20 31 29 41 64 64 20 2D 00 2D 20 32 29 53 75 62 74 72 61 63 74 20 2D 00 2D 20 33 29 4D 75 6C 74 69 70 6C 79 20 2D 00 2D 20 34 29 45 78 69 74 20 2D 00 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 0A 00 2D 20 31 29 41 64 64 20 2D 0A 00 2D 20 32 29 53 75 62 74 72 61 63 74 20 2D 0A 00 2D 20 33 29 4D 75 6C 74 69 70 6C 79 20 2D 0A 00 2D 20 34 29 45 78 69 74 20 2D 0A 00 20 2D 20 00 20 3D 20 00 00 00 00 00 00 00 00 00 00 00 47 43 43 3A 20 28 55 62 75 6E 74 75 20 37 2E 35 2E 30 2D 33 75 62 75 6E 74 75 31 7E 31 38 2E 30 34 29 20 37 2E 35 2E 30 00 00 00 00 00 00 00 14 00 00 00 00 00 00 00 01 7A 52 00 01 78 10 01 1B 0C 07 08 90 01 00 00 1C 00 00 00 1C 00 00 00 00 00 00 00 79 00 00 00 00 41 0E 10 86 02 43 0D 06 02 74 0C 07 08 00 00 1C 00 00 00 3C 00 00 00 00 00 00 00 AA 02 00 00 00 41 0E 10 86 02 43 0D 06 03 A5 02 0C 07 08 00 1C 00 00 00 5C 00 00 00 00 00 00 00 49 00 00 00 00 41 0E 10 86 02 43 0D 06 02 44 0C 07 08 00 00 1C 00 00 00 7C 00 00 00 00 00 00 00 15 00 00 00 00 41 0E 10 86 02 43 0D 06 50 0C 07 08 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 04 00 F1 FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 03 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 03 00 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 03 00 04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 03 00 05 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 13 00 00 00 01 00 05 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 2E 00 00 00 01 00 04 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 3D 00 00 00 02 00 01 00 23 03 00 00 00 00 00 00 49 00 00 00 00 00 00 00 6D 00 00 00 02 00 01 00 6C 03 00 00 00 00 00 00 15 00 00 00 00 00 00 00 00 00 00 00 03 00 06 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 03 00 09 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 03 00 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 03 00 08 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 7C 00 00 00 12 00 01 00 00 00 00 00 00 00 00 00 79 00 00 00 00 00 00 00 8D 00 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 97 00 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 AD 00 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 E5 00 00 00 12 00 01 00 79 00 00 00 00 00 00 00 AA 02 00 00 00 00 00 00 EA 00 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 F3 00 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 FE 00 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 08 01 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 43 01 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 54 01 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 65 01 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 7D 01 00 00 10 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 8A 01 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 A2 01 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 61 73 73 69 67 6E 6D 65 6E 74 36 5F 31 2E 63 70 70 00 5F 5A 53 74 4C 31 39 70 69 65 63 65 77 69 73 65 5F 63 6F 6E 73 74 72 75 63 74 00 5F 5A 53 74 4C 38 5F 5F 69 6F 69 6E 69 74 00 5F 5A 34 31 5F 5F 73 74 61 74 69 63 5F 69 6E 69 74 69 61 6C 69 7A 61 74 69 6F 6E 5F 61 6E 64 5F 64 65 73 74 72 75 63 74 69 6F 6E 5F 30 69 69 00 5F 47 4C 4F 42 41 4C 5F 5F 73 75 62 5F 49 5F 5F 5A 31 31 44 69 73 70 6C 61 79 4D 65 6E 75 76 00 5F 5A 53 74 34 63 6F 75 74 00 5F 47 4C 4F 42 41 4C 5F 4F 46 46 53 45 54 5F 54 41 42 4C 45 5F 00 5F 5A 53 74 6C 73 49 53 74 31 31 63 68 61 72 5F 74 72 61 69 74 73 49 63 45 45 52 53 74 31 33 62 61 73 69 63 5F 6F 73 74 72 65 61 6D 49 63 54 5F 45 53 35 5F 50 4B 63 00 6D 61 69 6E 00 5F 5A 53 74 33 63 69 6E 00 5F 5A 4E 53 69 72 73 45 52 69 00 5F 5A 4E 53 6F 6C 73 45 69 00 5F 5A 53 74 34 65 6E 64 6C 49 63 53 74 31 31 63 68 61 72 5F 74 72 61 69 74 73 49 63 45 45 52 53 74 31 33 62 61 73 69 63 5F 6F 73 74 72 65 61 6D 49 54 5F 54 30 5F 45 53 36 5F 00 5F 5A 4E 53 6F 6C 73 45 50 46 52 53 6F 53 5F 45 00 5F 5F 73 74 61 63 6B 5F 63 68 6B 5F 66 61 69 6C 00 5F 5A 4E 53 74 38 69 6F 73 5F 62 61 73 65 34 49 6E 69 74 43 31 45 76 00 5F 5F 64 73 6F 5F 68 61 6E 64 6C 65 00 5F 5A 4E 53 74 38 69 6F 73 5F 62 61 73 65 34 49 6E 69 74 44 31 45 76 00 5F 5F 63 78 61 5F 61 74 65 78 69 74 00 00 07 00 00 00 00 00 00 00 02 00 00 00 05 00 00 00 FD FF FF FF FF FF FF FF 0E 00 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF 13 00 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF 1A 00 00 00 00 00 00 00 02 00 00 00 05 00 00 00 0E 00 00 00 00 00 00 00 21 00 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF 26 00 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF 2D 00 00 00 00 00 00 00 02 00 00 00 05 00 00 00 18 00 00 00 00 00 00 00 34 00 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF 39 00 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF 40 00 00 00 00 00 00 00 02 00 00 00 05 00 00 00 27 00 00 00 00 00 00 00 47 00 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF 4C 00 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF 53 00 00 00 00 00 00 00 02 00 00 00 05 00 00 00 36 00 00 00 00 00 00 00 5A 00 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF 5F 00 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF 66 00 00 00 00 00 00 00 02 00 00 00 05 00 00 00 FD FF FF FF FF FF FF FF 6D 00 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF 72 00 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF A6 00 00 00 00 00 00 00 02 00 00 00 05 00 00 00 41 00 00 00 00 00 00 00 AD 00 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF B2 00 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF B9 00 00 00 00 00 00 00 02 00 00 00 05 00 00 00 53 00 00 00 00 00 00 00 C0 00 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF C5 00 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF CC 00 00 00 00 00 00 00 02 00 00 00 05 00 00 00 5E 00 00 00 00 00 00 00 D3 00 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF D8 00 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF DF 00 00 00 00 00 00 00 02 00 00 00 05 00 00 00 6E 00 00 00 00 00 00 00 E6 00 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF EB 00 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF F2 00 00 00 00 00 00 00 02 00 00 00 05 00 00 00 7E 00 00 00 00 00 00 00 F9 00 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF FE 00 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF 05 01 00 00 00 00 00 00 02 00 00 00 05 00 00 00 41 00 00 00 00 00 00 00 0C 01 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF 11 01 00 00 00 00 00 00 04 00 00 00 11

00 00 00 FC FF FF FF FF FF FF FF 1F 01 00 00 00 00 00 00 02 00 00 00 13 00 00 00 FC FF FF FF FF FF FF FF 24 01 00 00 00 00 00 00 04 00 00 00 14 00 00 00 FC FF FF FF FF FF FF FF 3E 01 00 00 00 00 00 00 02 00 00 00 13 00 00 00 FC FF FF FF FF FF FF FF 43 01 00 00 00 00 00 00 04 00 00 00 14 00 00 00 FC FF FF FF FF FF FF FF 55 01 00 00 00 00 00 00 04 00 00 00 14 00 00 00 FC FF FF FF FF FF FF FF 61 01 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF 66 01 00 00 00 00 00 00 04 00 00 00 15 00 00 00 FC FF FF FF FF FF FF FF 6D 01 00 00 00 00 00 00 02 00 00 00 05 00 00 00 8A 00 00 00 00 00 00 00 75 01 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF 85 01 00 00 00 00 00 00 04 00 00 00 15 00 00 00 FC FF FF FF FF FF FF FF 8C 01 00 00 00 00 00 00 02 00 00 00 05 00 00 00 8E 00 00 00 00 00 00 00 94 01 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF AB 01 00 00 00 00 00 00 04 00 00 00 15 00 00 00 FC FF FF FF FF FF FF FF B5 01 00 00 00 00 00 00 2A 00 00 00 16 00 00 00 FC FF FF FF FF FF FF FF C0 01 00 00 00 00 00 00 04 00 00 00 17 00 00 00 FC FF FF FF FF FF FF FF DF 01 00 00 00 00 00 00 02 00 00 00 13 00 00 00 FC FF FF FF FF FF FF FF E4 01 00 00 00 00 00 00 04 00 00 00 14 00 00 00 FC FF FF FF FF FF FF FF F6 01 00 00 00 00 00 00 04 00 00 00 14 00 00 00 FC FF FF FF FF FF FF FF 02 02 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF 07 02 00 00 00 00 00 00 04 00 00 00 15 00 00 00 FC FF FF FF FF FF FF FF 0E 02 00 00 00 00 00 00 02 00 00 00 05 00 00 00 8A 00 00 00 00 00 00 00 16 02 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF 26 02 00 00 00 00 00 00 04 00 00 00 15 00 00 00 FC FF FF FF FF FF FF FF 2D 02 00 00 00 00 00 00 02 00 00 00 05 00 00 00 8E 00 00 00 00 00 00 00 35 02 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF 4A 02 00 00 00 00 00 00 04 00 00 00 15 00 00 00 FC FF FF FF FF FF FF FF 54 02 00 00 00 00 00 00 2A 00 00 00 16 00 00 00 FC FF FF FF FF FF FF FF 5F 02 00 00 00 00 00 00 04 00 00 00 17 00 00 00 FC FF FF FF FF FF FF FF 7E 02 00 00 00 00 00 00 02 00 00 00 13 00 00 00 FC FF FF FF FF FF FF FF 83 02 00 00 00 00 00 00 04 00 00 00 14 00 00 00 FC FF FF FF FF FF FF FF 95 02 00 00 00 00 00 00 04 00 00 00 14 00 00 00 FC FF FF FF FF FF FF FF A1 02 00 00 00 00 00 00 02 00 00 00 0F 00 00 00 FC FF FF FF FF FF FF FF A6 02 00 00 00 00 00 00 04 00 00 00 15 00 00 00 FC FF FF FF FF FF FF FF AD 02 00 00 00 00 00 00 02 00 00 00 05 00 00 00 8A 00 00 00 00 00 00 00 B5 02 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF C5 02 00 00 00 00 00 00 04 00 00 00 15 00 00 00 FC FF FF FF FF FF FF FF CC 02 00 00 00 00 00 00 02 00 00 00 05 00 00 00 8E 00 00 00 00 00 00 00 D4 02 00 00 00 00 00 00 04 00 00 00 11 00 00 00 FC FF FF FF FF FF FF FF EA 02 00 00 00 00 00 00 04 00 00 00 15 00 00 00 FC FF FF FF FF FF FF FF F4 02 00 00 00 00 00 00 2A 00 00 00 16 00 00 00 FC FF FF FF FF FF FF FF FF 02 00 00 00 00 00 00 04 00 00 00 17 00 00 00 FC FF FF FF FF FF FF FF 1D 03 00 00 00 00 00 00 04 00 00 00 18 00 00 00 FC FF FF FF FF FF FF FF 43 03 00 00 00 00 00 00 02 00 00 00 04 00 00 00 FC FF FF FF FF FF FF FF 48 03 00 00 00 00 00 00 04 00 00 00 19 00 00 00 FC FF FF FF FF FF FF FF 4F 03 00 00 00 00 00 00 02 00 00 00 1A 00 00 00 FC FF FF FF FF FF FF FF 56 03 00 00 00 00 00 00 02 00 00 00 04 00 00 00 FC FF FF FF FF FF FF FF 5D 03 00 00 00 00 00 00 2A 00 00 00 1B 00 00 00 FC FF FF FF FF FF FF FF 65 03 00 00 00 00 00 00 04 00 00 00 1C 00 00 00 FC FF FF FF FF FF FF FF 00 00 00 00 00 00 00 00 01 00 00 00 02 00 00 00 6C 03 00 00 00 00 00 00 20 00 00 00 00 00 00 00 02 00 00 00 02 00 00 00 00 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00 02 00 00 00 02 00 00 00 79 00 00 00 00 00 00 00 60 00 00 00 00 00 00 00 02 00 00 00 02 00 00 00 23 03 00 00 00 00 00 00 80 00 00 00 00 00 00 00 02 00 00 00 02 00 00 00 6C 03 00 00 00 00 00 00 00 2E 73 79 6D 74 61 62 00 2E 73 74 72 74 61 62 00 2E 73 68 73 74 72 74 61 62 00 2E 72 65 6C 61 2E 74 65 78 74 00 2E 64 61 74 61 00 2E 62 73 73 00 2E 72 6F 64 61 74 61 00 2E 72 65 6C 61 2E 69 6E 69 74 5F 61 72 72 61 79 00 2E 63 6F 6D 6D 65 6E 74 00 2E 6E 6F 74 65 2E 47 4E 55 2D 73 74 61 63 6B 00 2E 72 65 6C 61 2E 65 68 5F 66 72 61 6D 65 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 20 00 00 00 01 00 00 00 06 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00 81 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 1B 00 00 00 04 00 00 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 90 09 00 00 00 00 00 00 E0 07 00 00 00 00 00 00 0C 00 00 00 01 00 00 00 08 00 00 00 00 00 00 00 18 00 00 00 00 00 00 00 26 00 00 00 01 00 00 00 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 C1 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 2C 00 00 00 08 00 00 00 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 C1 03 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 31 00 00 00 01 00 00 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 C1 03 00 00 00 00 00 00 96 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 3E 00 00 00 0E 00 00 00 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 58 04 00 00 00 00 00 00 08 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 08 00 00 00 00 00 00 00 08 00 00 00 00 00 00 00 39 00 00 00 04 00 00 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 70 11 00 00 00 00 00 00 18 00 00 00 00 00 00 00 0C 00 00 00 06 00 00 00 08 00 00 00 00 00 00 00 18 00 00 00 00 00 00 00 4A 00 00 00 01 00 00 00 30 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 60 04 00 00 00 00 00 00 2A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 53 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 8A 04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 68 00 00 00 01 00 00 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 90 04 00 00 00 00 00 00 98 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 08 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 63 00 00 00 04 00 00 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 88 11 00 00 00 00 00 00 60 00 00 00 00 00 00 00 0C 00 00 00 0A 00 00 00 08 00 00 00 00 00 00 00 18 00 00 00 00 00 00 00 01 00 00 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 28 05 00 00 00 00 00 00 B8 02 00 00 00 00 00 00 0D 00 00 00 0E 00 00 00 08 00 00 00 00 00 00 00 18 00 00 00 00 00 00 00 09 00 00 00 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 E0 07 00 00 00 00 00 00 AF 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 11 00 00 00 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 E8 11 00 00 00 00 00 00 72 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

**Step 4:** Convert the assembly code to C++ code.

| **Blocks of Assembly Code** | **C++ Code** |
| --- | --- |
| **push %rbp**  **mov %rsp,%rbp**  **lea 0x0(%rip),%rsi # 0xb <\_Z11DisplayMenuv+11>**  **lea 0x0(%rip),%rdi # 0x12 <\_Z11DisplayMenuv+18>**  **callq 0x17 <\_Z11DisplayMenuv+23>**  **lea 0x0(%rip),%rsi # 0x1e <\_Z11DisplayMenuv+30>**  **lea 0x0(%rip),%rdi # 0x25 <\_Z11DisplayMenuv+37>**  **callq 0x2a <\_Z11DisplayMenuv+42>**  **lea 0x0(%rip),%rsi # 0x31 <\_Z11DisplayMenuv+49>**  **lea 0x0(%rip),%rdi # 0x38 <\_Z11DisplayMenuv+56>**  **callq 0x3d <\_Z11DisplayMenuv+61>**  **lea 0x0(%rip),%rsi # 0x44 <\_Z11DisplayMenuv+68>**  **lea 0x0(%rip),%rdi # 0x4b <\_Z11DisplayMenuv+75>**  **callq 0x50 <\_Z11DisplayMenuv+80>**  **lea 0x0(%rip),%rsi # 0x57 <\_Z11DisplayMenuv+87>**  **lea 0x0(%rip),%rdi # 0x5e <\_Z11DisplayMenuv+94>**  **callq 0x63 <\_Z11DisplayMenuv+99>**  **lea 0x0(%rip),%rsi # 0x6a <\_Z11DisplayMenuv+106>**  **lea 0x0(%rip),%rdi # 0x71 <\_Z11DisplayMenuv+113>**  **callq 0x76 <\_Z11DisplayMenuv+118>**  **pop %rbp**  **retq** | void DisplayMenu() {  cout << "----------------" << endl;  cout << "- 1) Subtract -" << endl; //Originally Add  cout << "- 2) Add -" << endl; //Originally Subtract  cout << "- 3) Divide -" << endl; //Originally Multiply  cout << "- 4) Exit -" << endl;  cout << "----------------" << endl; |
| **mov -0x14(%rbp),%eax**  **cmp $0x5,%eax**  **je 0x308 <main+655>**  **lea 0x0(%rip),%rsi # 0xaa <main+49>**  **lea 0x0(%rip),%rdi # 0xb1 <main+56>**  **callq 0xb6 <main+61>** | int choice = 0;  DisplayMenu(); |
| **mov -0x14(%rbp),%eax**  **cmp $0x1,%eax**  **jne 0x1c9 <main+336>**  **lea -0x10(%rbp),%rax**  **mov %rax,%rsi**  **lea 0x0(%rip),%rdi # 0x142 <main+201>**  **callq 0x147 <main+206>**  **mov %rax,%rcx**  **mov -0x10(%rbp),%edx**  **mov -0xc(%rbp),%eax**  **sub %eax,%edx**  **mov %edx,%eax**  **mov %eax,%esi**  **mov %rcx,%rdi**  **callq 0x1af <main+310>**  **mov -0x14(%rbp),%eax**  **cmp $0x2,%eax**  **jne 0x268 <main+495>**  **lea -0x10(%rbp),%rax**  **mov %rax,%rsi**  **lea 0x0(%rip),%rdi # 0x1e3 <main+362>**  **callq 0x1e8 <main+367>**  **mov %rax,%rcx**  **mov -0x10(%rbp),%edx**  **mov -0xc(%rbp),%eax**  **add %edx,%eax**  **mov %eax,%esi**  **mov %rcx,%rdi**  **callq 0x24e <main+469>**  **mov -0x14(%rbp),%eax**  **cmp $0x3,%eax**  **jne 0x97 <main+30>**  **lea -0x10(%rbp),%rax**  **mov %rax,%rsi**  **lea 0x0(%rip),%rdi # 0x282 <main+521>**  **callq 0x287 <main+526>**  **mov %rax,%rcx**  **mov -0x10(%rbp),%eax**  **mov -0xc(%rbp),%esi**  **cltd**  **idiv %esi**  **mov %eax,%esi**  **mov %rcx,%rdi**  **callq 0x2ee <main+629>** | cin >> choice;  while (choice != 4) { //Change 5 to 4 to fit DisplayMenu  int num1, num2;    cout << "Please Enter First Number: " << endl;    cin >> num1;    cout << "Please Enter Second Number: " << endl;    cin >> num2;  if (choice == 1) {  cout << num1 << " - " << num2 << " = " << num1 - num2 << endl;  } else if (choice == 2) {  cout << num1 << " + " << num2 << " = " << num1 + num2 << endl; //Change " - " to " + " to fit output  } else if (choice == 3) {  cout << num1 << " / " << num2 << " = " << (double) num1 / (double) num2 << endl; //Change " - " to " / " to fit output  }  else  {    cout << "ERROR: Please Enter Valid Input" << endl; |
| **callq 0x321 <main+680>**  **leaveq**  **retq** | DisplayMenu();  cin >> choice;  }  } |