# **CS 410 Assembly to C++ Activity – Elliot Putnam**

Step 1: Convert the assembly code into C++ code.

Step 2: Explain the function of the converted C++ code.

| **Assembly Code** | **C++ Code** | **Explanation of Functionality** |
| --- | --- | --- |
| movl −8(%rbp), %eax sall $3, %eax subl $3, %eax movl %eax, −4(%rbp) | int eax = \*(rbp – 2);  eax <<= 3;  eax -= 3;  \*(rbp – 1) = eax; | load value from -8(%rbp)  shift left by 3 (multiply by 8)  subtract 3  store result at -4($rbp) |
| movl −8(%rbp), %eax sall $2, %eax subl $1, %eax leal 7(%rax), %edx testl %eax, %eax cmovs %edx, %eax sarl $3, %eax  movl %eax, −4(%rbp) | int eax = \*(rbp – 2);  eax <<= 2;  eax -= 1;  int edx = eax + 7;  if (eax == 0)  eax = edx;  eax >>= 3;  \*(rbp – 1) = eax; | load the value from -8(%rbp)  shift left by 2 (mult by 4)  subtract 1  store eax + 7 into eax  compare eax to 0  if eq, assign edx into eax  shift eax right by 3 (mult by 8)  store results in eax |
| movl −8(%rbp), %eax leal 7(%rax), %edx testl %eax, %eax cmovs %edx, %eax sarl $3, %eax movl −8(%rbp), %edx sall $2, %edx addl %edx, %eax  movl %eax, −4(%rbp) | int var;  int result = var + 7;  if (var >= 0)  var = result;  var = var / 8;  int var2;  var2 = var2 \* 4;  var = var + var2;  \*(int\*)(rbp – 4) = var; | load value into var  compute var+7 into var  if var >= 0  assign result to var  assign var / 8 to var  init var2  assign var2 \* 4 into var  assign var + var2 into var  store result of var at rbp -4 |