**2-2 Activity: Assembly to C++**

Dylan Jeffery

SNHU

CS-410

Brian West

1/16/2022

# **CS 410 Assembly to C++ Activity Template**

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 i;  int k = i \* 8 – 3; | movl initializes variable.  sall shifts the value of i, multiplying it by 8.  subl line subtracts 3 |
| 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 j;  int k = (j \* 4 -1) / 8; | movl initializes variable.  sall shifts the value of j, multiplying it by 4.  subl subtracts 1.  leal, testl, cmovs, and sarl perform division by 8. sarl performs the division but because it tends towards negative infinity the other lines ensure it behaves properly.  Final movl line places the value into memory |
| 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 a,k;  a = (a / 8) + (k \* 4); | movl initializes variable.  leal-sarl perform division by 8.  movl initializes another variable in another register.  sall shifts k to multiply by 4.  addl adds the value of each variable after the operations previously mentioned.  Save this value 4 bytes above %rbp |