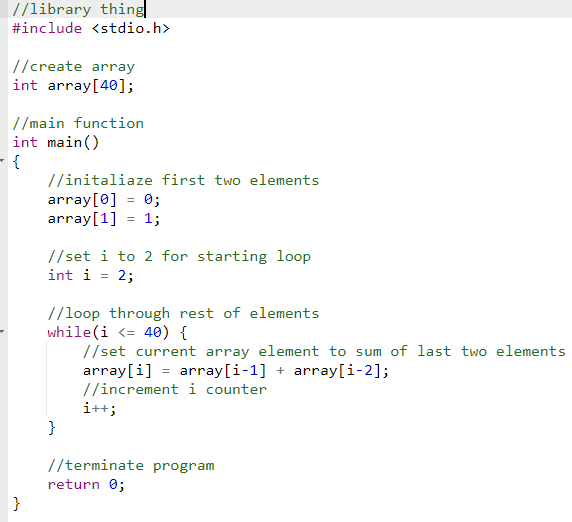
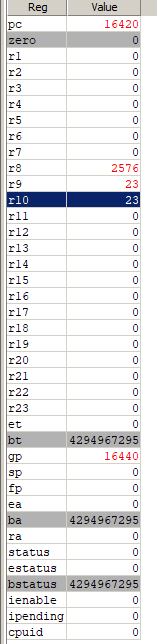
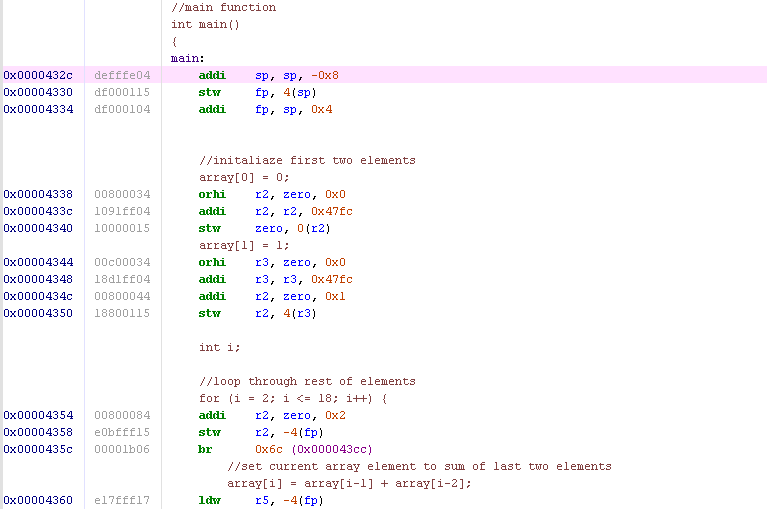
1. Pre-Lab (Pseudo code is included in the c code as comments)



8. Replacing “beq r9, r0, PROG\_END” with “beq r9, r10, PROG\_END” causes the program to stop not when it gets to the end of the array (which is marked with a 0) but instead to end when the program r9 or the current element is equal to 23. This causes the summation to be 2576, which is the value of r8, the current summation.



Part 2



1. a. 0x000043d8 - 0x0000432C = 0xAC = 172 bytes

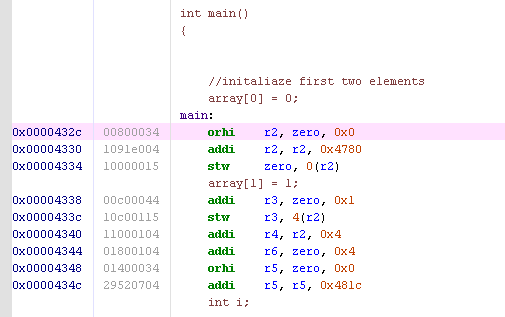
b. 172 / 4 = total # of instructions = 43

43 – loop (23) = non loop = 20

#loop\_instructions = 23 \* 37 = 851 loop instructions total

871 total instructions

c. (0x00004000 – 0x000043d8) = 984 bytes

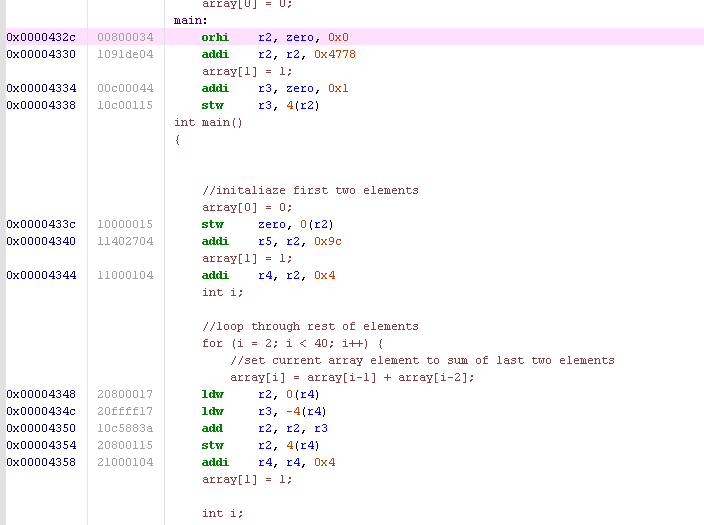


“-01”

3a. 15 instructions \* 4 = 60 bytes

b. 6\*23 + 9 = 147

c. (0x00004000-0x00004368) = 872 bytes

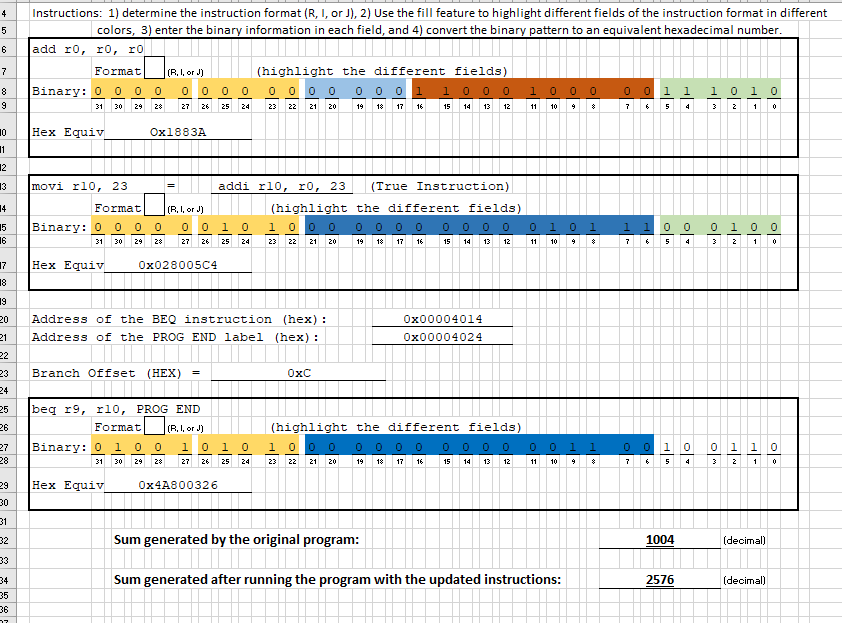


“-06”

3a. 13 instructions \* 4 = 52 bytes

b. 3 + 6 \* 37 = 225

c. (0x00004000 – 0x00004360) = 360 bytes



The effect of changing the optimization levels was that the size of the compiled code decreased as the optimization level increased. This is good because it allows for more programs to be stored, since each one will take up less space. This may be detrimental in that when you want to decompile code, it might be harder to do it when the code is more concise and probably harder to read.