# Name: \_\_\_\_\_\_\_\_\_\_

# Netid: \_\_\_\_\_\_\_\_\_\_\_

1. Implementation Tables for part 1 (correct design 55 pts). Copy the tables into this document and format using ‘Lucida Console’ font, size 9 so that the tables are readable. Or use a screenshot, just ensure that the tables are readable.
2. Implementation Tables for part 2 (correct design 20 pts). Copy the tables into this document and format using ‘Lucida Console’ font, size 9 so that the tables are readable. Or use a screenshot, just ensure that the tables are readable.
3. Question (5 pts): Estimate the number of LUTs (Lookup Tables) that your logic should generate. For each bit of a mux, use one LUT. For each adder bit, use two LUTs – one for the sum equation and one for the carry equation. For random logic like the Vflag, estimate based on the fact that one 6-input LUT can implement any logic equation of 6variables. Give a LUT total, and compare this to the actual number of LUTs generated by the synthesis tool. You will find that your estimate is somewhat higher than what the tool generated, we will discuss the reasons behind this in class. SHOW YOUR WORK TO GET CREDIT – give more than a single number!
4. Questions (20 pts):
   1. 0x610 + 0xA80 = \_\_\_\_ (normal hex addition), \_\_\_\_\_\_ (unsigned saturation), \_\_\_\_ (signed saturation)
   2. 0x704 + 0x2B0 = \_\_\_\_\_\_ (normal hex addition), \_\_\_\_\_\_ (unsigned saturation), \_\_\_\_\_ (signed saturation)
   3. 0x8F0 + 0xABF = \_\_\_\_\_\_ (normal hex addition), \_\_\_\_\_\_ (unsigned saturation), \_\_\_\_\_ (signed saturation)