Assume that you have one of these chips:
https://www.ti.com/lit/ds/symlink/cd74hc283.pdf

a) If A is 0100, B is 1101, and C_{in} is 0, what will be produced at S and C_{out}?

S will be 0001 and C_{out} will be 1.

b) If you have a 4-bit unsigned number N (carried on four wires named N_{4-0}) connected to both A and B, for what values of N is the expected result produced, assuming that C_{out} is not included as part of the output number?

The "expected result" is twice N (since N is being added to itself, i.e. N+N, i.e. 2*N).

The maximum number representable with four bits of unsigned binary is 15, so the expected result would be produced with N<=7.

c) In the same scenario as in (b), for what values of N is the expected result produced if C_{out} *is* included as part of the output number (forming a 5-bit output)?

The maximum number representable with five bits of unsigned binary is 31, so the expected result would be produced with N<=15, which is all possible N.

d) If you have a 4-bit two's complement number N (carried on four wires named N_{4-0}) connected to both A and B, for what values of N is the expected result produced, assuming that C_{out} is not included as part of the output number?

The addition algorithm for two's complement binary is the same as for unsigned binary, so this device works as we would want it to, at least as long as overflow doesn't occur. The range of numbers representable with four bits of two's complement binary is [-8,7], so the expected result would be produced with N in the range [-4,3].

e) In the same scenario as in (d), for what values of N is the expected result produced if C_{out} *is* included as part of the output number (forming a 5-bit output)?

This could be more complicated, because the two's complement number system is based on the modulo effect gained by restricting the number of places.

Positive number inputs can be up to 7 (0111), and 7+7 would result in 01110 – the correct result. So all positive numbers work.

Negative number inputs can be up to -8 (1000). -8+-8 would result in 10000 (-16) – also the correct result. So all negative numbers work as well.