Sélection internationale Session 2014
École Normale Supérieure Paris
Épreuve de culture scientifique - **Informatique** Durée : 2 heures

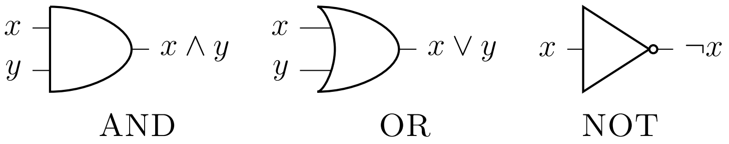
For candidates who chose **computer science** as **secondary specialisation**

If you cannot answer a question, you may use it as hypothesis to later questions.

Calculators are not allowed.

Exercice 1.

Logic gates are elements of a circuit with boolean input and outputs. The three (elementary) logic gates AND, OR and NOT are depicted below.



Please clearly mark all inputs and outputs in all circuit diagrams. Please refrain from using extra symbols as much as possible and explain what they represent in your answer.

1. Using only logic gates AND, OR and NOT, draw the circuit diagram for a 1-bit adder with input a_0, b_0 and (potentially) a carry r and output the sum s_0 of a_0, b_0 and r and carry s_1.
$$s_0 = a_0 + b_0 + r \bmod 2 \text{ and } s_1 = \lfloor (a_0 + b_0 + r)/2 \rfloor$$

2. Using only logic gates AND, OR, NOT and the circuit from the previous question, draw the circuit diagram from a 3-bit adder with inputs a_2, b_2, a_1, b_1, a_0, b_0, r representing two 3-bit numbers a and b in binary (written $a_2a_1a_0$ and $b_2b_1b_0$), and a one bit number r. It should output the sum s of these three numbers in binary as s_3, s_2, s_1, s_0.
$$\sum_{i=0}^{3} s_i 2^i = \sum_{i=0}^{2} a_i 2^i + \sum_{i=0}^{2} b_i 2^i + r$$
Explain how to extend this to an n-bit adder.

3. Determine the number of gates $g(n)$ needed for an n-bit adder in the circuit from Question 2.

4. Suppose it takes t seconds for a signal propagate through an elementary gate. If all input signals are altered on an n-bit adder from Question 2 at time 0, what is the time needed for all output signal to be propagated (as a function of n and t).

We say a 2^k*-bit divide-and-conquer adder* is a circuit obtained recursively as follows:

- The 1-bit divide-and-conquer adder is the 1-bit adder from Question 1
- For $k \geq 1$, the 2^k-bit divide-and-conquer adder is obtained by joining two 2^{k-1}-bit divide-and-conquer adders A and B by sending the lower 2^{k-1} of (the inputs) a and b as well as the carry r to A and all remaining bits to B. Then the output of the carry of A is sent to B.

5. Deduce the delay from inputs to outputs for a 2^k-bit divide-and-conquer.

We now define an *improve 2^k-bit divide-and-conquer*. Here, the left copy of the divide-and-conquer adder will be replaced by two adders, one supposing the input carry is 1 and another supposing the input carry is 0.

6. Draw the circuit diagram using three 2^{k-1}-bit improved divide-and-conquers and some elementary logic gates to produce a 2^k-bit improved divide-and-conquer adder.

7. Deduce the delay from inputs to outputs for a 2^k-bit improved divide-and-conquer.

8. Derive a recursive formula to deduce the number of elementary logic gates in a 2^k-bit improved divide-and-conquer.