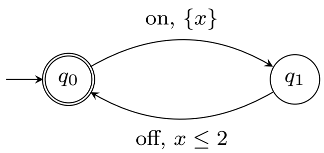
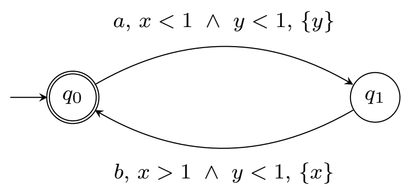
1. Consider an automaton with 2 clocks $\{x, y\}$. Let the maximum bounds function M for the automaton be given by: $M(x) = 3, M(y) = 4$. Draw the division of the xy-plane into regions.

2. Given 3 clocks $\{x, y, z\}$ and $M(x) = 3, M(y) = 4, M(z) = 2$, enumerate the set of regions.

3. Let R be a region over clock set X and bound function M. Give an algorithm to compute the time-successors of a region R.

4. Draw the region graph for the following automata:





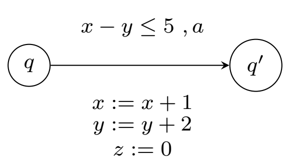
5. Suppose R is a region over clock set X and bound function M. Let x, y be two arbitrary clocks in X. Is the projection of R on to the xy-plane a region over $\{x, y\}$ with the bounds function M restricted to x and y?

6. Let us add an extra feature to the timed automaton model. Suppose in addition to resets that set a clock to 0, we also allow resetting a clock to 1: that is, each transition is of the form (q, a, g, R_0, R_1, q') where g is the guard, R_0 is the set of clocks that have to be reset to 0 and R_1 is the set of clocks that need to be reset to 1 (assume that $R_0 \cap R_1 = \emptyset$ in every transition).

   Let TA_{+1} denote the set of timed automata that have these special resets to either 0 or 1.

   Show that this extra feature does not add expressive power to the model. In other words, prove that for every automaton $\mathcal{A}$ in TA_{+1} there exists a normal timed automaton $\mathcal{B}$ that has resets only to 0, such that $\mathcal{L}(\mathcal{A}) = \mathcal{L}(\mathcal{B})$.

7. Consider timed automata *with diagonal constraints*. Suppose we extend this model to allow more complicated resets of the form: $x := x + c$ where x is a clock and c is a natural number, in addition to the traditional resets that assign a subset of clocks to 0. For example, the following figure shows a transition in such an automaton:



   The transition from q to q' has the diagonal guard $x - y \leq 5$. Once the transition is taken, the value of x is increased by 1 from its current value, the value of y is increased by 2 and the value of z is set to 0 (the normal reset). More formally, each transition is of the form (q, a, g, R, q') where R is a function that maps each clock x to either 0 or $x + c$, where $c \in \mathbb{N}$.

   Let $\text{TA}^d_{x:=x+c}$ denote the set of timed automata that can have diagonal guards and the special resets described above. Show that the following language can be recognized by a timed automaton in $\text{TA}^d_{x:=x+c}$:

   $$\{ \ (w, \tau) \mid w \in (a + b)^*, \tau \text{ is some time sequence, and } w \text{ has the same number of } a\text{'s and } b\text{'s} \}$$

   Can the above language be recognized by a normal timed automaton which has resets only to 0?

8. Prove that the language emptiness problem for the class of timed automata $\mathrm{TA}^d_{x:=x+c}$ described in the above question is undecidable.

   You may use the following undecidable problem.

   A Minsky machine (a version of 2-counter machine) consists of a finite set of labeled instructions $I_1, \ldots, I_n$ and two counters c_1, c_2. There is a specified initial instruction I_0 and a special instruction labeled HALT. The instructions are of two types:

   - an *incrementation* instruction of counter $c \in \{c_1, c_2\}$

$$p: \ c := c + 1; \text{ goto } q \qquad \text{(where } p, q \text{ are instruction labels)}$$

   - or a *decrementation (or zero-testing)* instruction of counter $c \in \{c_1, c_2\}$

$$p: \ \text{if } c > 0 \begin{cases} \text{then } c := c - 1; \text{ goto } q \\ \text{else goto } r \end{cases} \qquad \text{(where } p, q, r \text{ are instruction labels)}$$

   The machine starts at instruction I_0 with counters $c_1 = c_2 = 0$, executes the instructions successively, and stops only when it reaches the instruction HALT. The halting problem for Minsky machine is to decide if there is an execution of the machine that reaches the instruction HALT.

   It is known that the halting problem for Minsky machines is undecidable.