A finite set

Problem: BinOpGen

Input: A finite set X, a binary operation \( * : X \times X \rightarrow X \), a subset \( S \subseteq X \) and a target \( t \in X \).

Question: Does \( \langle S \rangle \) contain \( t \)?

Here \( \langle S \rangle \) denotes the closure of \( S \), a set of generating elements, by \( * \). Formally, we define an increasing sequence \( S_0, S_1, S_2, \ldots \) of subsets of \( X \) with

\[
S_0 \overset{\text{def}}{=} S, \quad \forall i \in \mathbb{N} : S_{i+1} \overset{\text{def}}{=} S_i \cup \{ x * y \mid x, y \in S_i \},
\]

and we let \( \langle S \rangle \overset{\text{def}}{=} \bigcup_{i \in \mathbb{N}} S_i \).

For representation purposes, we may assume that \( X \) is a finite set of the form \( \{1, \ldots, n\} \), so that \( * \) can be represented as an \( n \times n \) matrix with values in \( \{1, \ldots, n\} \).

A related problem is 2BinOpGen, where we consider the closure of \( S \) via two binary operations on \( X \), say \( *_1 \) and \( *_2 \). Formally, we define \( \langle S \rangle_{*_1,*_2} \overset{\text{def}}{=} \bigcup_{i \in \mathbb{N}} S_i, *_{_1,*_2} \) replacing (1) with

\[
S_{i+1}, *_{_1,*_2} = S_i, *_{_1,*_2} \cup \{ x *_{_1} y \mid x, y \in S_i \} \cup \{ x *_{_2} y \mid x, y \in S_i \}
\]

and again asking whether \( t \in \langle S \rangle_{*_{_1,*_2}} \).

Question 1

Show that BinOpGen and 2BinOpGen are inter-reducible, hence “have the same complexity”.

Solution:

The reduction BinOpGen \( \leq \) 2BinOpGen is easy: we duplicate \( * \) and leave \( X \), \( S \), \( t \) unchanged.

For 2BinOpGen \( \leq \) BinOpGen, a possible solution is to define a reduction

\[
X, *_{_1,*_2}, S, t \mapsto X', *', S', t'
\]

with \( X' \overset{\text{def}}{=} X \times \{1, 2\} \) and \( *', S', t' \) given by

\[
(x, k) *' (y, l) \overset{\text{def}}{=} (x *_k y, l), \quad S' \overset{\text{def}}{=} S \times \{1, 2\}, \quad t' \overset{\text{def}}{=} (t, 1).
\]

In other words, \( X' \) is made of two copies of \( X \). We claim that \( S_i', *' = S_i, *_{_1,*_2} \times \{1, 2\} \) for all \( i \in \mathbb{N} \) and prove this by induction on \( i \) (easy proof omitted). This entails \( t \in \langle S \rangle_{*_{_1,*_2}} \) iff \( (t, 1) \in \langle S' \rangle_{*'} \), so the reduction is correct. That it is logspace is clear since outputting \( *' \) just needs two nested loops on \( X \), and \( X' \), \( S' \) and \( t' \) are even easier to produce.

TernOpGen is a further variant where we are given a ternary operation, denoted \( \phi \), over \( X \), and where one asks, given \( X, \phi, S, t \), whether \( t \in \langle S \rangle_{\phi} \). Here (1) is replaced by \( S_{i+1,\phi} \overset{\text{def}}{=} S_i, \phi \cup \{ \phi(x, y, z) \mid x, y, z \in S_i, \phi \} \).

Question 2

Show that BinOpGen and TernOpGen are inter-reducible.
Solution:
The reduction witnessing $\text{BinOpGen} \leq \text{TernOpGen}$ can be as simple as $X, \ast, S, t \mapsto X, \phi, S, t$ with $
abla x, y, z \nabla \equiv x \ast y$. We then have $x \in S_{i, \ast}$ iff $x \in S_{i, \phi}$ for all $x \in X$ and $i \in \mathbb{N}$, hence the reduction is correct.

For proving $\text{TernOpGen} \leq \text{BinOpGen}$, one possible solution is to define a reduction

$$X, \phi, S, t \mapsto X', \ast, S, t$$

with $X' = X \cup X^2$ (assuming that $X$ and $X^2$ are disjoint). In other words, $\ast$ operates on elements from $X$, but also on pairs from $X^2$. For all $x, y, z, u \in X$, we let

$$x \ast y \equiv (x, y), \quad x \ast (y, z) \equiv \phi(x, y, z), \quad (x, y) \ast (z, u) \equiv (x, y) \ast z \equiv (x, y).$$

We claim that $S_{i, \ast} \subseteq S_{2i, \ast}$ for all $i \in \mathbb{N}$, and also that $S_{i, \ast} \subseteq S_{i, \phi} \cup (S_{i, \phi} \times S_{i, \phi})$ for all $i \in \mathbb{N}$. Each claim is easily proven by induction on $i$. Put together they entail $t \in \langle S \rangle_\ast$ iff $t \in \langle S \rangle_\phi$, hence the reduction is correct. It is logspace since producing $X'$ and $\ast$ requires two and four nested loops on $X$.

Let $\text{BinOpGen}_{\text{assoc}}$ be the restriction of $\text{BinOpGen}$ to the case where $\ast$ is associative, i.e., satisfies $x \ast (y \ast z) = (x \ast y) \ast z$ for all $x, y, z \in X$.

It will be useful to recall the definition of the Graph Accessibility Problem from the Dec. 9th class:

**Problem : GAP**

**Input :** A finite directed graph $G = (V, E)$, with $E \subseteq V \times V$, and two vertices $s, t \in V$.

**Question :** Is there a path $s = v_0 \rightarrow v_1 \rightarrow \cdots \rightarrow v_t = t$ from $s$ to $t$, written $s \stackrel{*}{\rightarrow} t$, in $G$?

**Question 3**

Show that $\text{GAP} \leq \text{BinOpGen}_{\text{assoc}}$.

**Solution:**
We consider a reduction

$$G = (V, E), s, t \mapsto X, \ast, S, t'$$

where $X \equiv V^2 \cup \{\perp\}$, $t' \equiv (s, t)$, and with $\ast$ and $S$ given by :

$$(v_1, v_2) \ast (v_3, v_4) = \begin{cases} (v_1, v_4) & \text{if } v_2 = v_3, \\ (v_1, \perp) & \text{otherwise}, \end{cases} \quad \perp \ast x = x \ast \perp = \perp, \quad S \equiv E \cup \text{Id}_V.$$

Observe that $\ast$ is indeed associative (with $\perp$ as absorbing element), as required for $\text{BinOpGen}_{\text{assoc}}$.

We now claim that, for all $v, v' \in V$ and $i \in \mathbb{N}$, $(v, v') \in S_{i, \ast}$ iff $G$ has a path $v \stackrel{*}{\rightarrow} v'$ of length $\leq 2^i$. This is easily proven by induction on $i$. As a consequence, $t' = (s, t) \in \langle S \rangle_\ast$ iff $G$ has a path $s \stackrel{*}{\rightarrow} t$, hence the reduction is correct.

The reduction is logspace since producing $\ast$ only requires four nested loops on $V$, while $X, S$ and $t'$ are even easier to produce.

**Question 4**

Show that $\text{BinOpGen}_{\text{assoc}} \leq \text{GAP}$.

**Solution:**
We consider a reduction

$$X, \ast, S, t \mapsto (V, E), s, t$$

where $V = \{0\} \cup X \cup X^2$ (assuming these 3 sets are disjoint), $s \equiv 0$, and where the edges are given by $E = E_1 \cup E_2 \cup E_3$ with

$${E_1 \equiv \{(0, x) \mid x \in S\} \cup (x, y) \mid x \ast y \in X \}, \quad {E_2 \equiv \{(x, y) \mid x \ast y \in X\} \cup \{(x, (x, y)) \mid x \in X, y \in S\} \}}.$$

Let us now argue that the reduction is correct:

If $t \in \langle S \rangle_\ast$, then, and since $\ast$ is associative, we can write $t = x_1 \ast x_2 \cdots \ast x_t$ for some $x_1, \ldots, x_t \in S$.

Writing $p_i$ for the partial product $x_1 \ast x_2 \ast \ldots \ast x_i$, we see that $G$ has a path

$$s = 0 \xrightarrow{E_1} x_1 \xrightarrow{E_2} (x_1, x_2) \xrightarrow{E_3} x_1 \ast x_2 = p_2 \xrightarrow{E_1} (p_2, x_3) \xrightarrow{E_2} p_3 \xrightarrow{E_3} \cdots \xrightarrow{E_3} (p_{t-1}, x_t) \xrightarrow{E_2} p_t = t.$$
Reciprocally, and since $G$ is bipartite, any path $s = 0 \to t$ with $t \in V$ must have the form

$$0 \overset{E_1}{\rightarrow} x_1 \overset{E_2}{\rightarrow} (x_1, y_2) \overset{E_3}{\rightarrow} x_2 \overset{E_4}{\rightarrow} (x_2, y_3) \overset{\cdots}{\rightarrow} E_m (x_{m-1}, y_m) \overset{E_3}{\rightarrow} x_m = t.$$ 

The definitions of $E_1, E_2, E_3$ enforce $x_1, y_2, y_3, \ldots, y_m \in S$ and $x_{i+1} = x_i \ast y_{i+1}$ for all $i = 1, \ldots, m-1$, hence $x_m = x_1 \ast y_2 \ast \cdots \ast y_m$, showing $t \in \langle S \rangle_+$. 