Presentations of semigroups, groups and other algebraic structures

Robert D. Gray

LMS Summer Graduate Student Meeting
London, June 2019

University of East Anglia

Presentations in generators and relations

Presentations are a fundamental tool for describing algebraic objects as homomorphic images of free objects.

Good news: Presentations allow us to define certain infinite structures with a finite amount of data.

- Finitely presented groups, semigroups, rings etc.

Bad news: In general it is difficult (sometimes impossible) to say much about an algebraic structure defined by a finite presentation.

- The word problem, and other decision problems

Classical versus non-classical algebra

Classical	Non classical
Groups, rings, k-algebras (with k a field)	Semigroups, monoids, lattices, universal algebras
homomorphisms and substructures	homomorphisms and substructures
normal subgroups, ideals	congruences
Quotient G/N, R/I	Quotients S/ρ with ρ a congruence

- I will mainly focus on the theory of presentations for semigroups, monoids and groups, with some indication of how ideas extend to arbitrary algebraic structures.
- The theory for semigroups & monoids will give a flavour of concepts that arise in the general theory of presentations of universal algebras.

Words

A – a non-empty set called an alphabet (e.g. $A = \{a, b\}$)

$A^* = \{$all words written in the letters from $A\}$
(A^* also contains the empty word ϵ, which is the word with no letters.)

If $u, v \in A^*$ we can multiply them together by concatenating them to obtain a new word $uv \in A^*$.

Example

$abbab, baa \in A^*$ and their product is the word $abbabbaa$.

Definition

Let A be a non-empty set. The set A^* together with the operation of concatenation of words is called the free monoid on the alphabet A.

$A^+ = \{$all non-empty words over the alphabet$\}$
This is the free semigroup on the alphabet A.

Semigroups, monoids and groups

Definition

A semigroup is a pair $(S, \cdot)$ where S is a non-empty set and $\cdot$ is a binary operation defined on S that satisfies the associative law

$$(x \cdot y) \cdot z = x \cdot (y \cdot z)$$

for all $x, y, z \in S$. The product of two elements x and y is usually written just as xy rather than $x \cdot y$.

If a semigroup contains an element 1 with the property that $x1 = 1x = x$ for all $x \in S$ then we call 1 the identity element of the S and we call S a monoid.

Clearly we have the containment of classes:

$$\text{Groups} \subseteq \text{Monoids} \subseteq \text{Semigroups}$$

Presentations

$$\langle A \mid R \rangle = \langle \quad \underbrace{a_1, \ \ldots, \ a_n}_{\text{letters / generators}} \quad \mid \quad \underbrace{u_1 = v_1, \ \ldots, \ u_m = v_m}_{\text{words / defining relations}} \quad \rangle$$

- Defines the monoid $S = A^*/\rho$ where ρ is the smallest congruence on the free monoid A^* containing R.
- S is the free-est / largest (in terms of homomorphic images) monoid generated by A in which the generators satisfy all the relations R.

How to think about S

- Elements of S are equivalence classes of words over A.
- Two words are in the same equivalence class (i.e. they represent the same element of S) if one can be transformed into the other by applying the relations R.

Definition: S is finitely presented if both A and R finite.

Presentations

Example: $S \cong \langle A \mid R \rangle = \langle a, b \mid ab = ba \rangle$

Words $u, v \in A^*$ represent the same element of S if u can be transformed into v by a finite number of applications of the relations.

$$\text{e.g.} \quad abaa = aaba = aaab, \qquad abb \neq aab.$$

Fact: Every word $u \in A^*$ is equal in S to a unique word of the form $a^i b^j$, and these normal forms multiply as

$$(a^{i_1} b^{j_1})(a^{i_2} b^{j_2}) = a^{i_1 + i_2} b^{j_1 + j_2}.$$

Presentations

Example: $S \cong \langle A \mid R \rangle = \langle a, b \mid ab = ba \rangle$

Words $u, v \in A^*$ represent the same element of S if u can be transformed into v by a finite number of applications of the relations.

$$\text{e.g.} \quad abaa = aaba = aaab, \qquad abb \neq aab.$$

Fact: Every word $u \in A^*$ is equal in S to a unique word of the form $a^i b^j$, and these normal forms multiply as

$$(a^{i_1} b^{j_1})(a^{i_2} b^{j_2}) = a^{i_1+i_2} b^{j_1+j_2}.$$

The word problem: For any $u, v \in \{a, b\}^*$ we have

$$\begin{aligned} u = v \iff \quad & u \text{ and } v \text{ have the same number occurrences of the letter } a \\ \& \quad & u \text{ and } v \text{ have the same number occurrences of the letter } b. \end{aligned}$$

The word problem

Definition

A monoid S with a finite generating set A has decidable word problem if there is an algorithm which for any two words $w_1, w_2 \in A^*$ decides whether or not they represent the same element of S.

Example. $S \cong \langle a, b \mid ab = ba \rangle$ has decidable word problem.

Example

Let M be the monoid defined by the presentation:

$$\mathcal{P} = \langle A \mid R \rangle = \left\langle x, y \mid y^2 = y, xy = y \right\rangle .$$

- Do we have $xy^3xyx = x^2yxy^2x$ in the monoid M?

The word problem

Definition

A monoid S with a finite generating set A has decidable word problem if there is an algorithm which for any two words $w_1, w_2 \in A^*$ decides whether or not they represent the same element of S.

Example. $S \cong \langle a, b \mid ab = ba \rangle$ has decidable word problem.

Example

Let M be the monoid defined by the presentation:

$$\mathcal{P} = \langle A \mid R \rangle = \left\langle x, y \mid y^2 = y, xy = y \right\rangle .$$

- Do we have $xy^3xyx = x^2yxy^2x$ in the monoid M?
- Do we have $yx^2y = xy^2x^2$ in the monoid M?

The word problem

Definition

A monoid S with a finite generating set A has decidable word problem if there is an algorithm which for any two words $w_1, w_2 \in A^*$ decides whether or not they represent the same element of S.

Example. $S \cong \langle a, b \mid ab = ba \rangle$ has decidable word problem.

Example

Let M be the monoid defined by the presentation:

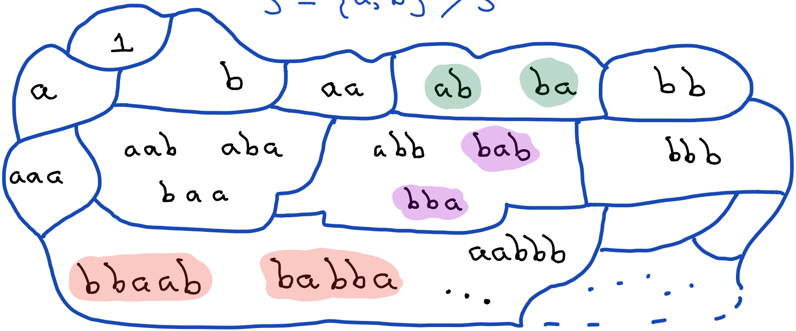
$$\mathcal{P} = \langle A \mid R \rangle = \left\langle x, y \mid y^2 = y, xy = y \right\rangle .$$

- Do we have $xy^3xyx = x^2yxy^2x$ in the monoid M?
- Do we have $yx^2y = xy^2x^2$ in the monoid M?
- If we suspect $yx^2y \neq xy^2x^2$ how do we prove it?

Monoid defined by a presentation is a quotient of the free monoid

$S \cong \langle a, b \mid ab = ba \rangle$, ρ = smallest congruence containing (ab, ba)

$$S = \{a, b\}^*/\rho$$



$$(bba/\rho)(ab/\rho) = bbaab/\rho = babba/\rho = (bab/\rho)(ba/\rho)$$

<u>Multiplication</u> $(w_1/\rho)(w_2/\rho) = w_1 w_2/\rho$, for $w_1, w_2 \in A^*$

Congruences and quotient semigroups

Definition: $\langle A \mid R \rangle \cong A^*/\rho$ where ρ is the smallest congruence on A^* containing R.

Definition

An equivalence relation ρ on a semigroup S is a congruence if it is compatible with multiplication, i.e.

$$(x, y), (z, t) \in \rho \Rightarrow (xz, yt) \in \rho \text{ for all } x, y, z, t \in S.$$

Theorem

Let ρ be a congruence on a semigroup S. The quotient semigroup S/ρ has elements the set of equivalence classes and multiplication defined by

$$(x/\rho)(y/\rho) = (xy)/\rho.$$

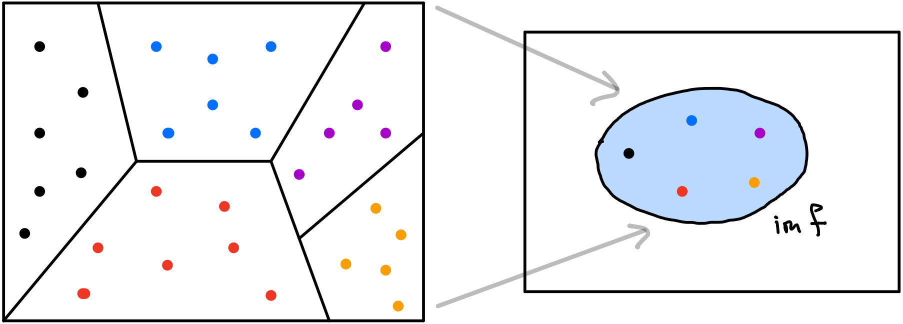
- Groups: congruences determined by normal subgroups giving G/N.
- Rings: congruences are determined by ideals giving R/I.

First isomorphism theorem: There is a correspondence between:

$$\text{Quotients} \longleftrightarrow \text{Homomorphic images}$$

First isomorphism theorem

$$M \xrightarrow[\text{homomorphism}]{f} S$$



$$\rho = \ker f = \{ (x,y) \in M \times M : f(x) = f(y) \}$$

is a congruence and

$$M/\rho = M/\ker f \cong \operatorname{im} f$$

Universal property of free monoids

Definition: $\langle A \mid R \rangle \cong A^*/\rho$ where ρ is the smallest congruence on A^* containing R.

Theorem

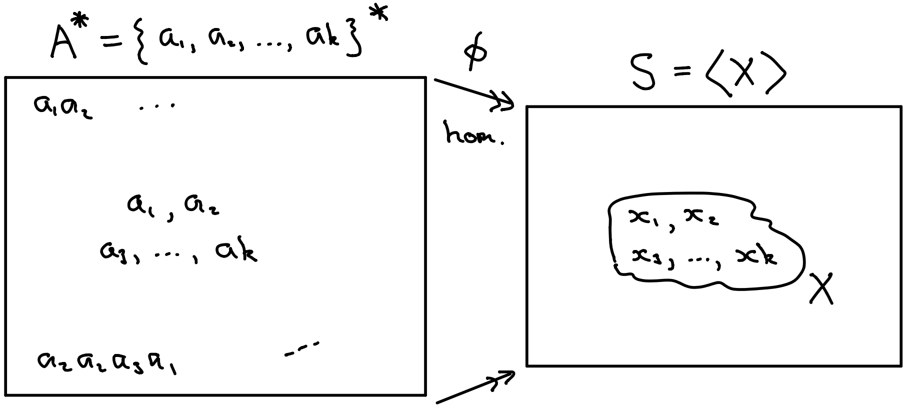
Let S be a monoid and let let A be an alphabet. Then any mapping $f : A \to S$ extends uniquely to a homomorphism $\phi : A^* \longrightarrow S$.

Therefore, every monoid is a homomorphic image of a free monoid.

Equivalently, every monoid is a quotient of some free monoid.

This implies that every monoid is defined by some (not necessarily finite) presentation.

Every monoid is homomorphic image of free



The mapping $a_i \longmapsto x_i$ extends to the homomorphism

$$\phi : a_{i_1} a_{i_2} \cdots a_{i_m} \longrightarrow x_{i_1} x_{i_2} \cdots x_{i_m} .$$

The free group

Definition

The free group on A is defined as

$$\mathrm{FG}(A) = \left\langle A \cup A^{-1} \mid aa^{-1} = 1,\ a^{-1}a = 1\ (a \in A) \right\rangle$$

- Elements of $\mathrm{FG}(A) \longleftrightarrow$ Reduced words

where

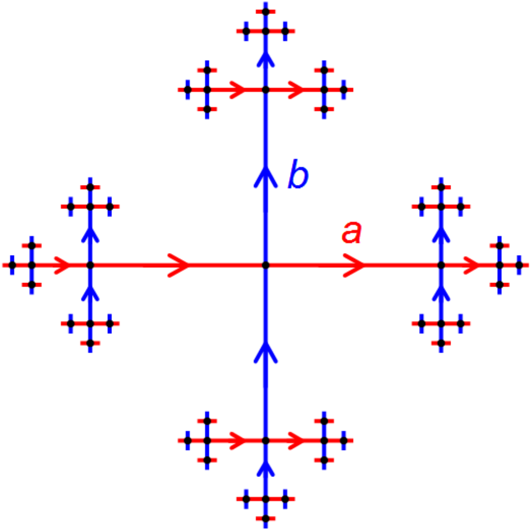
$$w \text{ is reduced} \iff w \text{ does not contain } aa^{-1} \text{ or } a^{-1}a.$$

Example

$aba^{-1}b^{-1}$ and $baab$ are reduced words, and their product is

$$(aba^{-1}b^{-1})(baab) = abab.$$

Elements of the free group $\mathrm{FG}(a, b)$



Group presentations

G - a group generated by A,   for any $u, v \in (A \cup A^{-1})^*$

$$u = v \text{ in } G \Leftrightarrow uv^{-1} = 1 \text{ in } G.$$

Therefore, any relation can be written in the form $r = 1$.

Definition

Let R be a subset of $\mathrm{FG}(A)$, so $R \subseteq (A \cup A^{-1})^*$ is a set of reduced words. Then we define

$$\mathrm{Gp}\langle A \mid R\rangle = \left\langle A \cup A^{-1} \mid aa^{-1} = 1,\ a^{-1}a = 1\ (a \in A),\ r = 1\ (r \in R)\right\rangle$$

and call this the group defined by the presentation $\mathrm{Gp}\langle A \mid R\rangle$ with generators A and defining relations R.

Example

$\mathrm{Gp}\langle a \mid a^3 = 1\rangle$ defines the cyclic group of order three.

Proposition

$\mathrm{Gp}\langle A \mid R\rangle \cong \mathrm{FG}(A)/\langle\!\langle R\rangle\!\rangle$ where $\langle\!\langle R\rangle\!\rangle$ is the normal closure of R in $\mathrm{FG}(A)$.

Word problem as sets of words

- The word problem for the monoid M defined by the presentation $\langle A \mid R\rangle$ can be viewed as the set

$$\{(w_1, w_2) \in A^* \times A^* : w_1 = w_2 \text{ in the monoid } M\}.$$

- The word problem for the group $G \cong \mathrm{Gp}\langle A \mid R\rangle$ is the set

$$\{w \in \mathrm{FG}(A) : w = 1 \text{ in the group } G\}.$$

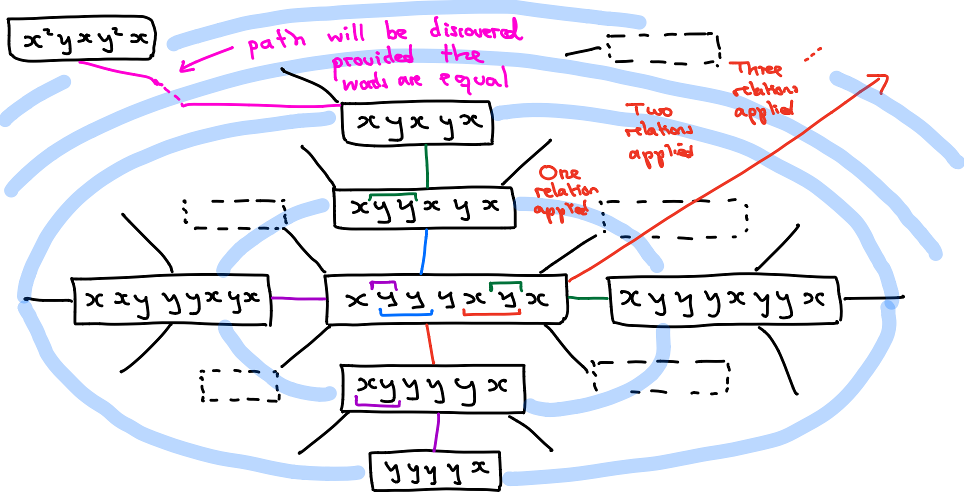
Computing subsets of A^*

- A set W of words is computably enumerable if there is an algorithm which takes any word u as input and, if u is a member of W, then the algorithm eventually halts and says YES; otherwise it runs forever.
- A set W of words is computable if there is an algorithm which takes any word u as input, terminates after a finite amount of time and decides whether or not the word u belongs to W, returning either YES or NO.

Important fact: The word problem for any finitely presented semigroup, monoid, or group, is a computably enumerable set.

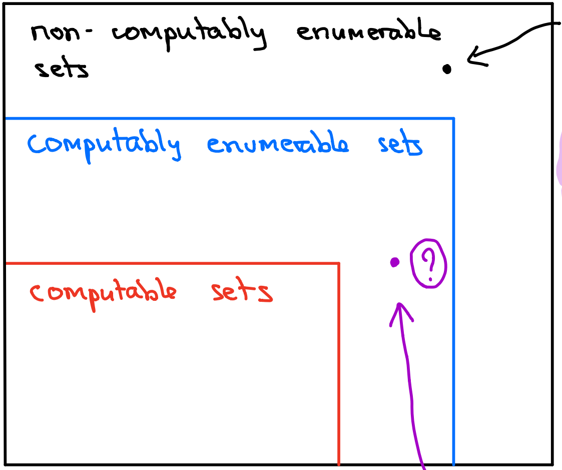
$M = \langle x, y \mid y^2 = y \, , \, xy = y \rangle$ To discover that $xy^3xyx = x^2yxy^2x$ in M, for each i compute the set $B_i = \{ w \in \{x,y\}^* : w$ can be obtained from xy^3xyx by applying $\leq i$ defining relations $\}$

Then check does $x^2yxy^2x \in B_i$?



A^* – free monoid over finite alphabet A.

$\mathcal{P}(A^*) = \{$ subsets of $A^* \}$ – Power set



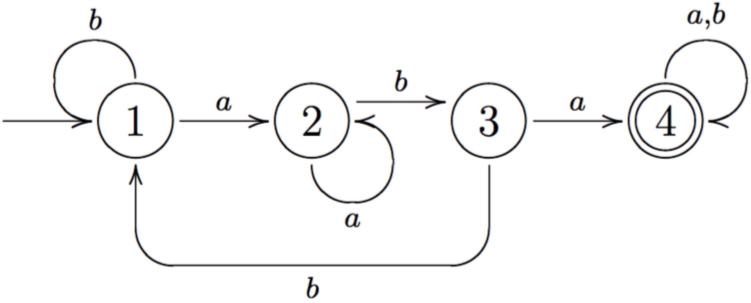
∃ uncountably many sets here since $\mathcal{P}(A^*)$ is uncountable and there are only countably many algorithms & so only countably many computably enumerable sets.

∴ It is obvious that there are non-c.e. (& hence non-computable) sets.

To construct finitely presented groups/semigroups with undecidable word problem we need to show there are sets here

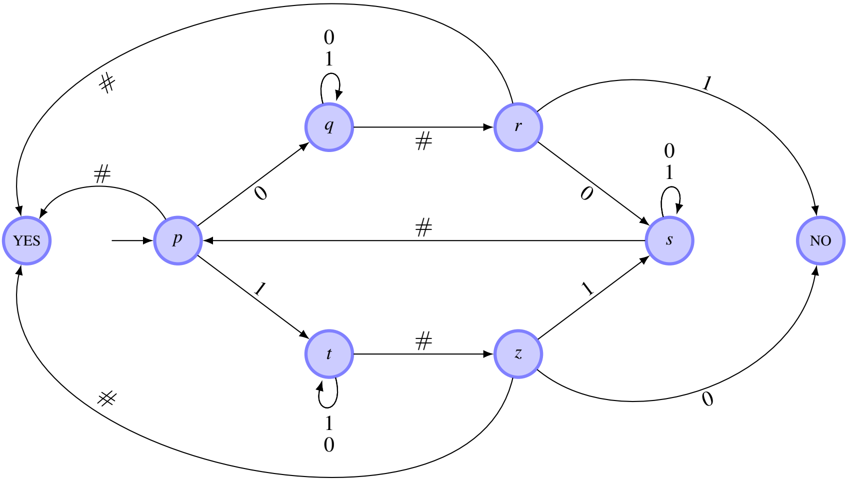
<u>Key question</u> Are there c.e. sets that are not computable ?

Finite state automata

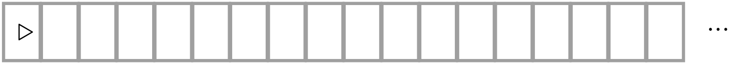


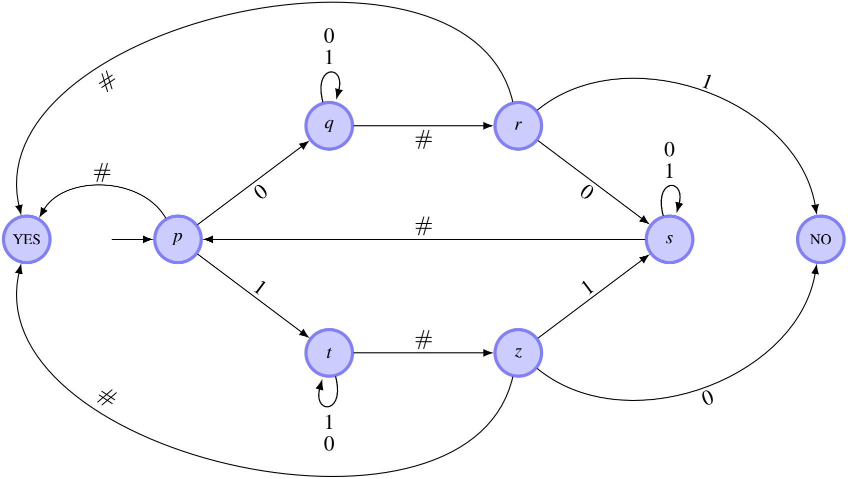
- Alphabet: $A = \{a, b\}$.
- $L(\mathcal{A}) \subseteq A^*$ - language of words recognised by the automaton $\mathcal{A}$
  - e.g. here $aba \in L(\mathcal{A})$ while $abba \notin L(\mathcal{A})$
- Languages recognised by finite state automata are the regular languages.
- Every regular language is computable, but there are computable languages that are not regular.

Turing machines, alphabet $A = \{0, 1\}$



Turing machines, alphabet $A = \{0, 1\}$

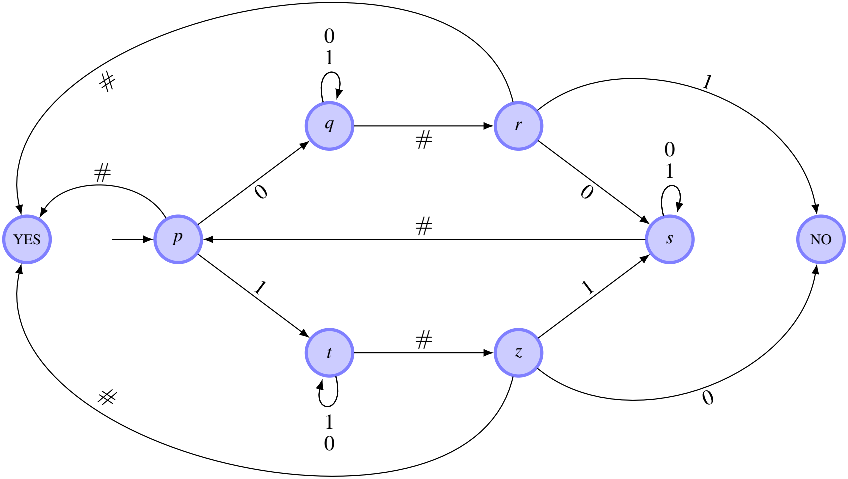




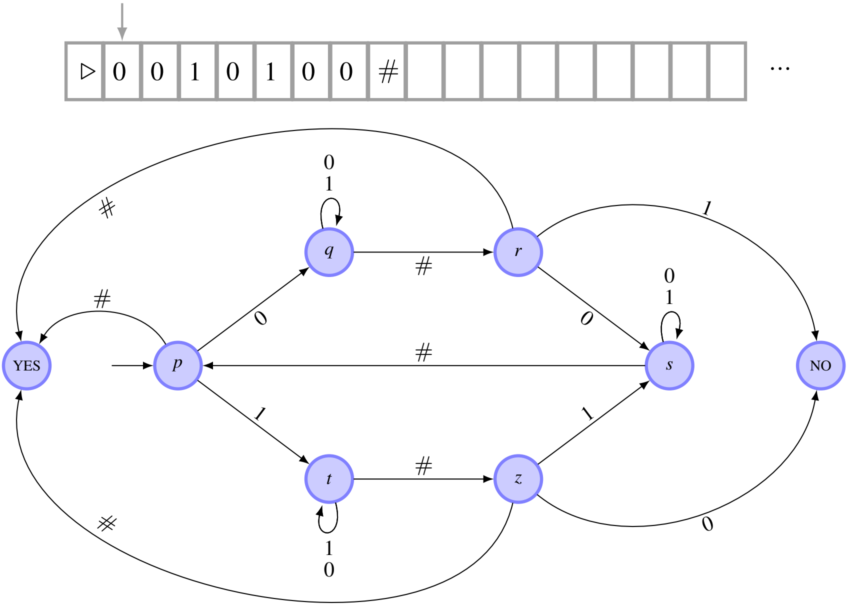
Turing machines, alphabet $A = \{0, 1\}$

▷	0	0	1	0	1	0	0	#									

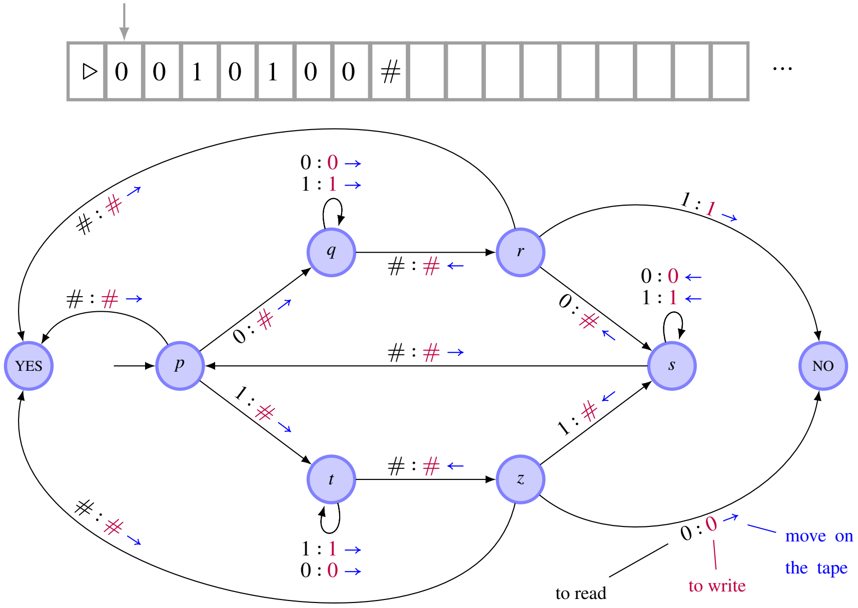
…



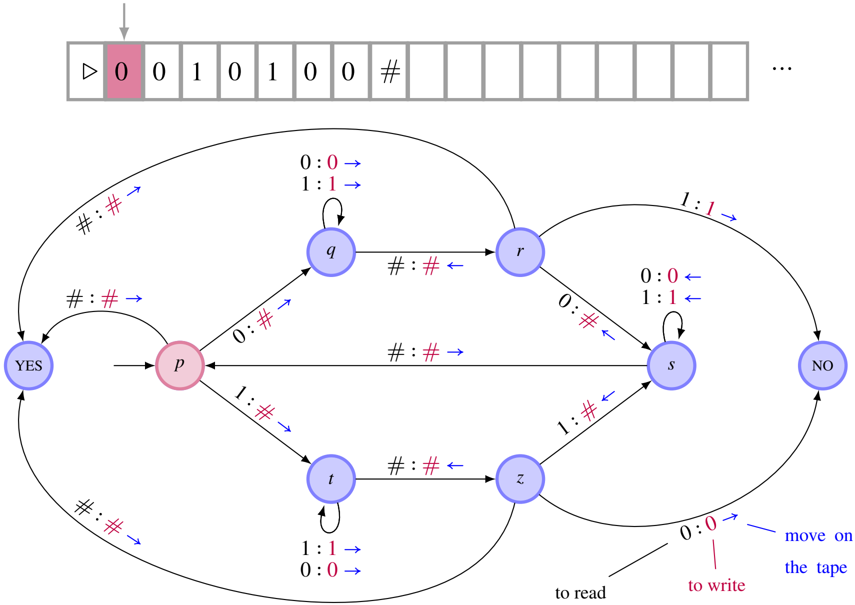
Turing machines, alphabet $A = \{0, 1\}$



Turing machines, alphabet $A = \{0, 1\}$



Turing machines, alphabet $A = \{0, 1\}$

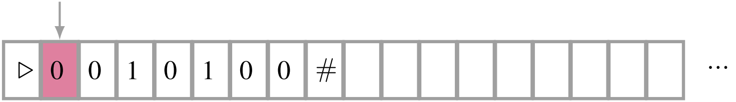


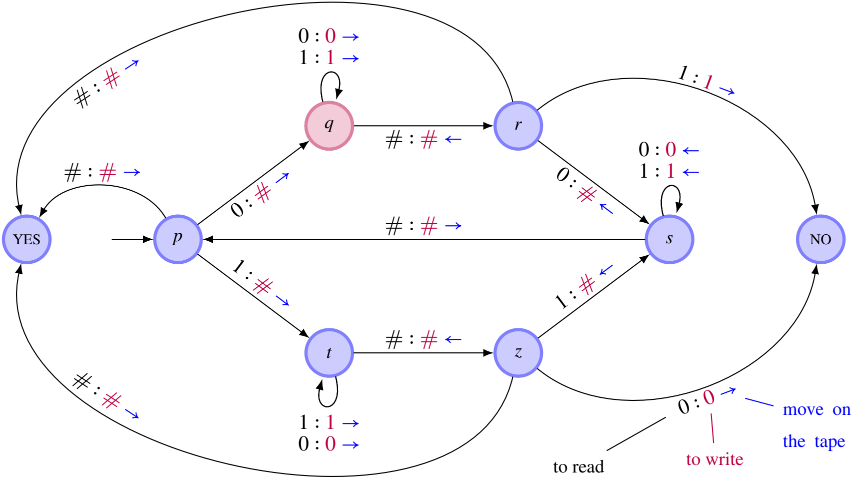
to read

to write

move on the tape

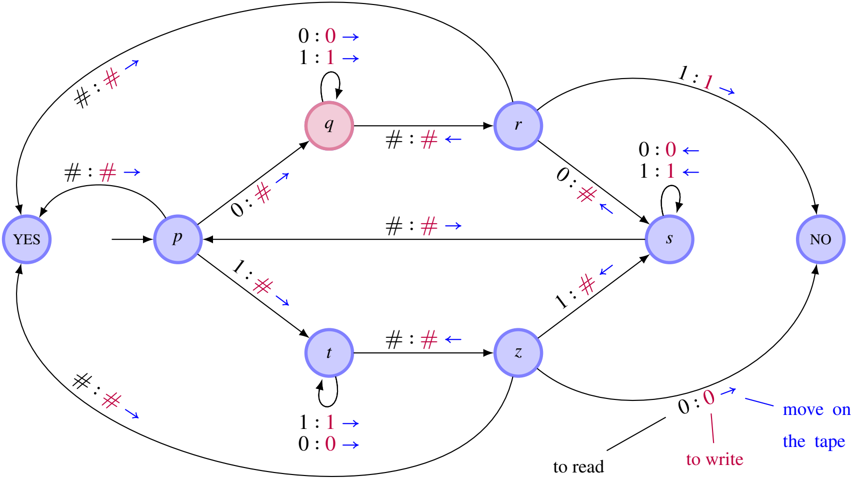
Turing machines, alphabet $A = \{0, 1\}$





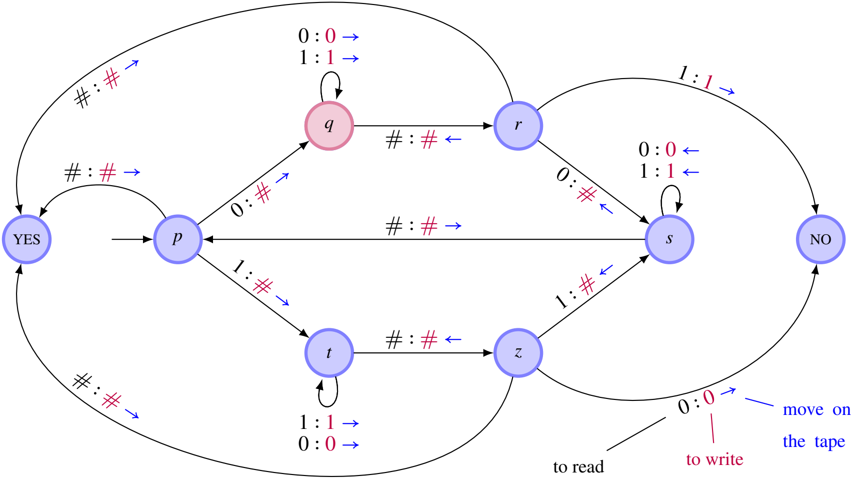
Turing machines, alphabet $A = \{0, 1\}$

▷	#	0	1	0	1	0	0	#										…

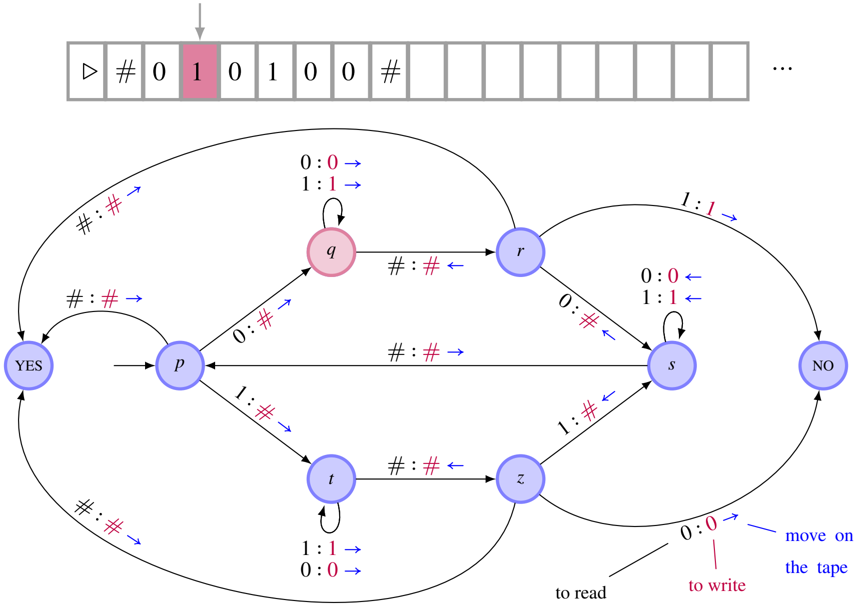


Turing machines, alphabet $A = \{0, 1\}$

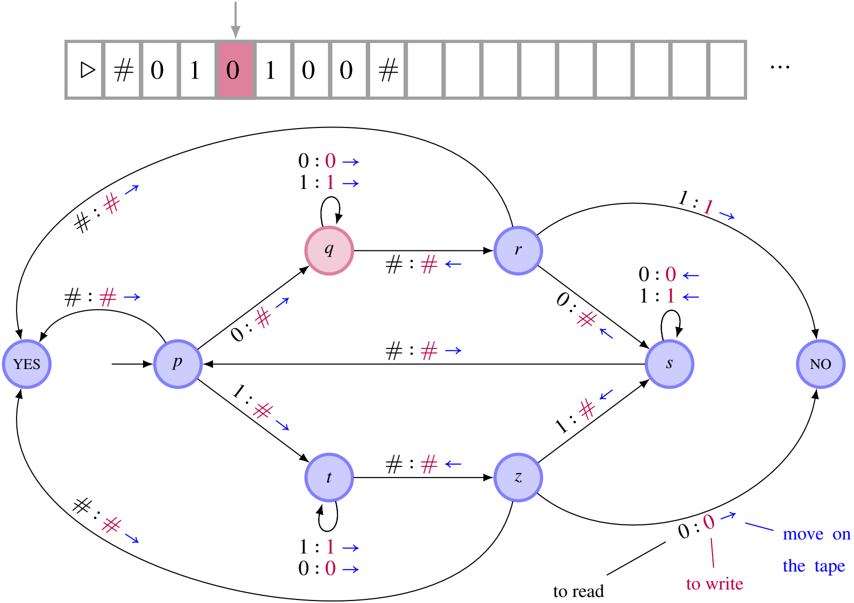
▷	#	0	1	0	1	0	0	#										…



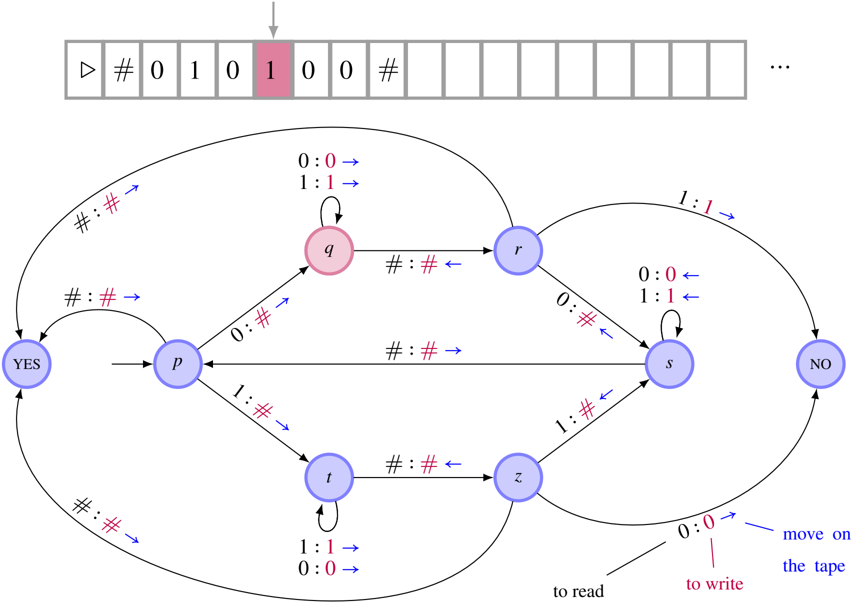
Turing machines, alphabet $A = \{0, 1\}$



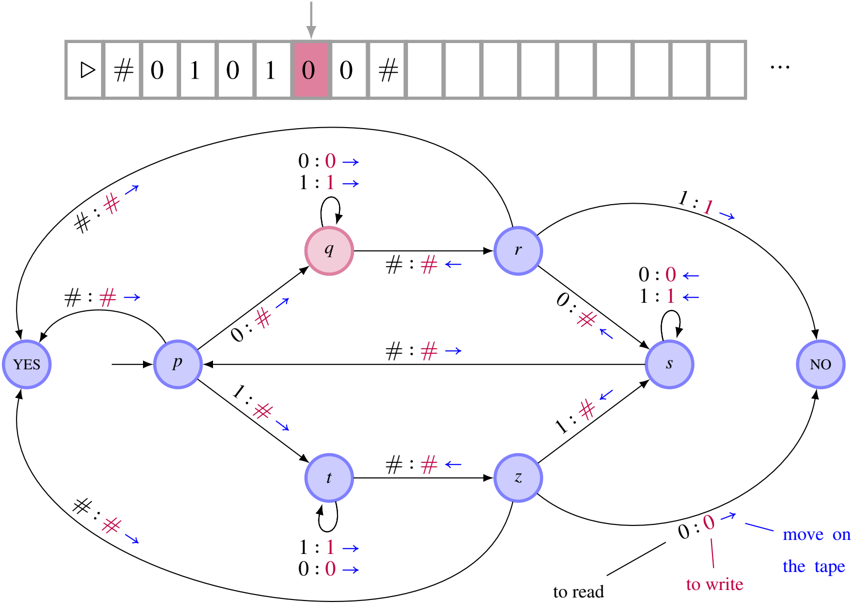
Turing machines, alphabet $A = \{0, 1\}$



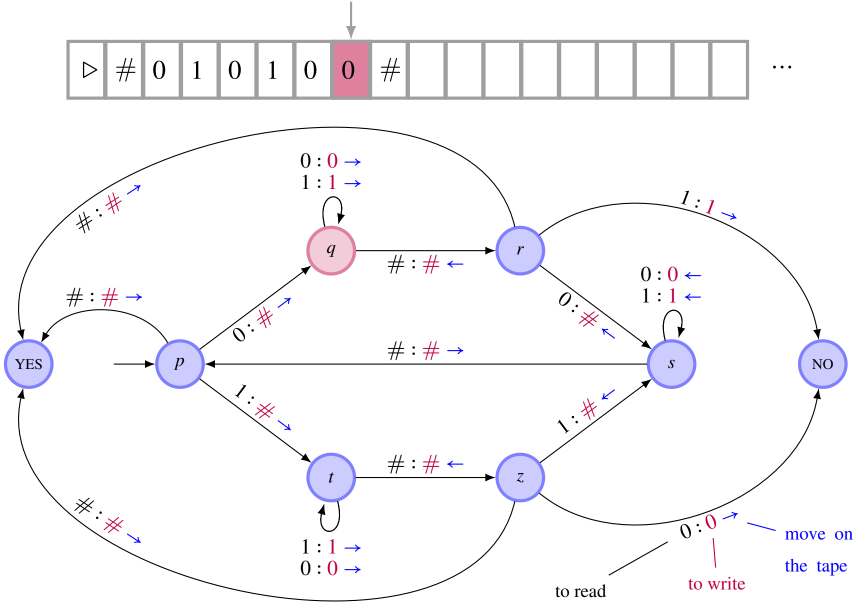
Turing machines, alphabet $A = \{0, 1\}$



Turing machines, alphabet $A = \{0, 1\}$



Turing machines, alphabet $A = \{0, 1\}$

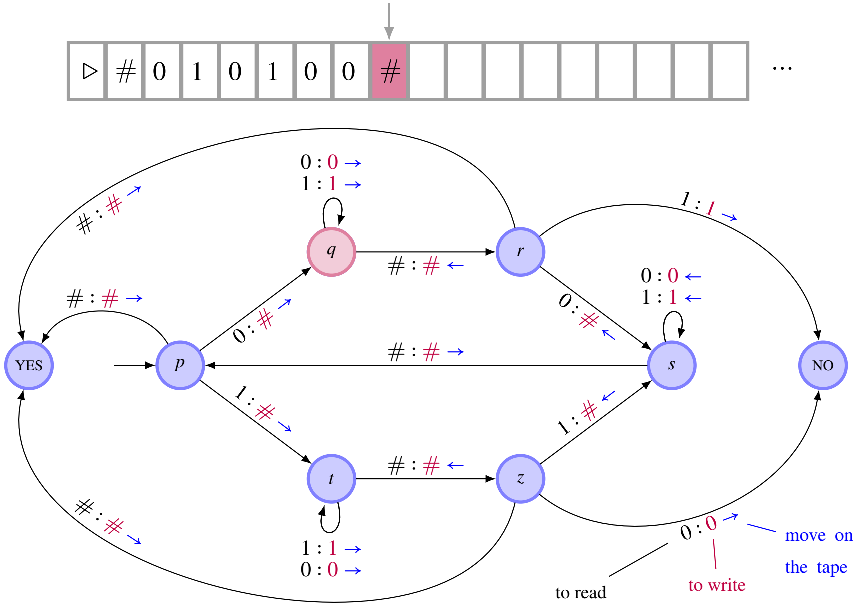


to read

to write

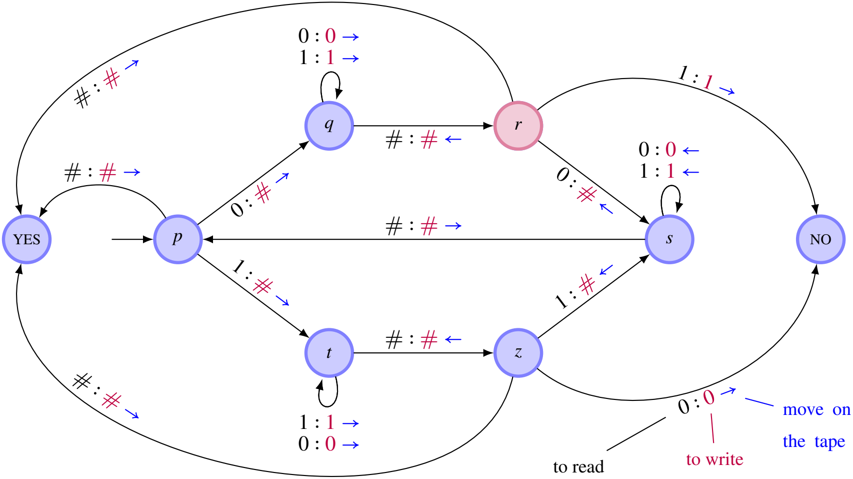
move on the tape

Turing machines, alphabet $A = \{0, 1\}$

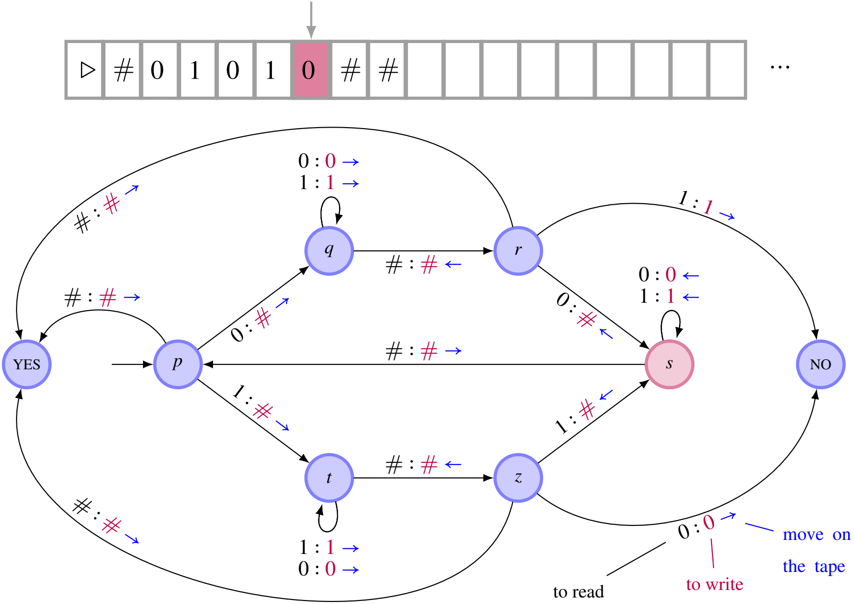


Turing machines, alphabet $A = \{0, 1\}$

▷	#	0	1	0	1	0	0	#										…



Turing machines, alphabet $A = \{0, 1\}$



Russell’s paradox

The village barber shaves exactly those men in the village who don’t shave themselves.

Does the barber shave himself, or not?

Undecidable problems

P - some computer program. Try entering P into itself. Either:

The program will run forever... – immortal program
It terminates at some point – suicidal program

The halting problem

Does there exist a program $\mathcal{P}$ that can test whether a program is immortal or suicidal?

Undecidable problems

P - some computer program. Try entering P into itself. Either:

The program will run forever... – immortal program
It terminates at some point – suicidal program

The halting problem

Does there exist a program $\mathcal{P}$ that can test whether a program is immortal or suicidal?

If yes, modify $\mathcal{P}$ so that it
– goes into a loop when given a suicidal program as input, and
– terminates given an immortal program as input.

Undecidable problems

P - some computer program. Try entering P into itself. Either:

The program will run forever... – immortal program
It terminates at some point – suicidal program

The halting problem

Does there exist a program $\mathcal{P}$ that can test whether a program is immortal or suicidal?

If yes, modify $\mathcal{P}$ so that it
– goes into a loop when given a suicidal program as input, and
– terminates given an immortal program as input.

Q: Is $\mathcal{P}$ itself an immortal program or a suicidal program...?

Undecidable problems

P - some computer program. Try entering P into itself. Either:

The program will run forever... – immortal program
It terminates at some point – suicidal program

The halting problem

Does there exist a program $\mathcal{P}$ that can test whether a program is immortal or suicidal?

If yes, modify $\mathcal{P}$ so that it
– goes into a loop when given a suicidal program as input, and
– terminates given an immortal program as input.

Q: Is $\mathcal{P}$ itself an immortal program or a suicidal program...?

Conclusion (Turing (1936)): The halting problem is undecidable.

Computably enumerable but not computable sets

The halting set

$$\mathcal{T} = \{(p, i) \mid \text{the program } p \text{ halts when run on input } i\}$$

represents the halting problem. The halting set can be used to show the existence of subsets of $\mathbb{N}$ (and of A^*, $FG(A)$, etc...) that are computably enumerable, but are not computable.

This underlies the following fundamental results:

- Markov (1947), Post (1947): There exist finitely presented semigroups with undecidable word problem.
- Novikov (1955) and Boone (1958): There exist finitely presented groups with undecidable word problem.

Céjtin (1957): The monoid with presentation $\langle A \mid R \rangle$ where $A = \{a, b, c, d, e\}$ and defining relations

$$ac = ca,\ ad = da,\ bc = cb,\ bd = db,\ ce = eca,\ de = edb,\ cca = ccae$$

has undecidable word problem.

Residual finiteness and the word problem

Definition

A semigroup S is residually finite if for any two distinct elements $x, y \in S$ there is a homomorphism $\phi : S \to T$ onto a finite semigroup T such that $x\phi \neq y\phi$.

Examples

The infinite cyclic group $(\mathbb{Z}, +)$.
The free semigroup A^+ and free monoid A^*.

On the other hand the bicyclic monoid $\langle b, c \mid bc = 1 \rangle$ is not residually finite.

Residual finiteness and the word problem

Theorem (Malcev's lemma)

Let S be a finitely presented semigroup. If S is residually finite then S has a decidable word problem.

Note: The same results holds for finitely presented groups, monoids, semigroups, ... etc.

Proof of Theorem (Malcev's lemma)

Let $S \cong \langle A \mid R \rangle$ finitely presented and residually finite. Let $u, v \in A^+$. Set two programs P_{yes} and P_{no} running in parallel:

Proof of Theorem (Malcev's lemma)

Let $S \cong \langle A \mid R \rangle$ finitely presented and residually finite. Let $u, v \in A^+$. Set two programs P_{yes} and P_{no} running in parallel:

(P_{yes}) Begin with the set $B_0 = \{u\}$ and then at step $i \in \mathbb{N}$ compute

$B_i = B_{i-1} \cup \{w \in A^+ : \exists w' \in B_{i-1}$ such that w can be obtained from w' by a single application of a relation from $R\}$.

Proof of Theorem (Malcev's lemma)

Let $S \cong \langle A \mid R \rangle$ finitely presented and residually finite. Let $u, v \in A^+$. Set two programs P_{yes} and P_{no} running in parallel:

(P_{yes}) Begin with the set $B_0 = \{u\}$ and then at step $i \in \mathbb{N}$ compute

$B_i = B_{i-1} \cup \{w \in A^+ : \exists w' \in B_{i-1}$ such that w can be obtained from w' by a single application of a relation from $R\}$.

Check if $v \in B_i$. If yes, stop and output YES, otherwise move to step $i + 1$.

Proof of Theorem (Malcev's lemma)

Let $S \cong \langle A \mid R \rangle$ finitely presented and residually finite. Let $u, v \in A^+$. Set two programs P_{yes} and P_{no} running in parallel:

(P_{yes}) Begin with the set $B_0 = \{u\}$ and then at step $i \in \mathbb{N}$ compute

$$B_i = B_{i-1} \cup \{w \in A^+ : \exists w' \in B_{i-1} \text{ such that } w \text{ can be obtained from } w' \text{ by a single application of a relation from } R\}.$$

Check if $v \in B_i$. If yes, stop and output YES, otherwise move to step $i + 1$.

(P_{no}) At step i, write down multiplication tables of all finite semigroups of size i. For each finite semigroup T of size i and for every mapping $f : A \to T$ check whether $\overline{f}(l) = \overline{f}(r)$ in T for each of the defining relations $(l, r) \in R$, where $\overline{f}$ is the unique extension of f to a homomorphism $\overline{f} : A^+ \to T$.

Proof of Theorem (Malcev's lemma)

Let $S \cong \langle A \mid R \rangle$ finitely presented and residually finite. Let $u, v \in A^+$. Set two programs P_{yes} and P_{no} running in parallel:

(P_{yes}) Begin with the set $B_0 = \{u\}$ and then at step $i \in \mathbb{N}$ compute

$B_i = B_{i-1} \cup \{w \in A^+ : \exists w' \in B_{i-1}$ such that w can be obtained from w' by a single application of a relation from $R\}$.

Check if $v \in B_i$. If yes, stop and output YES, otherwise move to step $i + 1$.

(P_{no}) At step i, write down multiplication tables of all finite semigroups of size i. For each finite semigroup T of size i and for every mapping $f : A \to T$ check whether $\overline{f}(l) = \overline{f}(r)$ in T for each of the defining relations $(l, r) \in R$, where $\overline{f}$ is the unique extension of f to a homomorphism $\overline{f} : A^+ \to T$. If yes, then check whether or not $\overline{f}(u) = \overline{f}(v)$ in T. If $\overline{f}(u) \neq \overline{f}(v)$ in T then stop and output NO, otherwise move to step $i + 1$.

Proof of Theorem (Malcev's lemma)

Let $S \cong \langle A \mid R \rangle$ finitely presented and residually finite. Let $u, v \in A^+$. Set two programs P_{yes} and P_{no} running in parallel:

(P_{yes}) Begin with the set $B_0 = \{u\}$ and then at step $i \in \mathbb{N}$ compute

$$B_i = B_{i-1} \cup \{w \in A^+ : \exists w' \in B_{i-1} \text{ such that } w \text{ can be obtained from } w' \text{ by a single application of a relation from } R\}.$$

Check if $v \in B_i$. If yes, stop and output YES, otherwise move to step $i + 1$.

(P_{no}) At step i, write down multiplication tables of all finite semigroups of size i. For each finite semigroup T of size i and for every mapping $f : A \to T$ check whether $\overline{f}(l) = \overline{f}(r)$ in T for each of the defining relations $(l, r) \in R$, where $\overline{f}$ is the unique extension of f to a homomorphism $\overline{f} : A^+ \to T$. If yes, then check whether or not $\overline{f}(u) = \overline{f}(v)$ in T. If $\overline{f}(u) \neq \overline{f}(v)$ in T then stop and output NO, otherwise move to step $i + 1$.

$u = v$ in $S \Leftrightarrow$ (P_{yes}) eventually terminates with output YES.

Proof of Theorem (Malcev's lemma)

Let $S \cong \langle A \mid R \rangle$ finitely presented and residually finite. Let $u, v \in A^+$. Set two programs P_{yes} and P_{no} running in parallel:

(P_{yes}) Begin with the set $B_0 = \{u\}$ and then at step $i \in \mathbb{N}$ compute

$B_i = B_{i-1} \cup \{w \in A^+ : \exists w' \in B_{i-1}$ such that w can be obtained from w' by a single application of a relation from $R\}$.

Check if $v \in B_i$. If yes, stop and output YES, otherwise move to step $i + 1$.

(P_{no}) At step i, write down multiplication tables of all finite semigroups of size i. For each finite semigroup T of size i and for every mapping $f : A \to T$ check whether $\overline{f}(l) = \overline{f}(r)$ in T for each of the defining relations $(l, r) \in R$, where $\overline{f}$ is the unique extension of f to a homomorphism $\overline{f} : A^+ \to T$. If yes, then check whether or not $\overline{f}(u) = \overline{f}(v)$ in T. If $\overline{f}(u) \neq \overline{f}(v)$ in T then stop and output NO, otherwise move to step $i + 1$.

$u = v$ in $S \Leftrightarrow$ (P_{yes}) eventually terminates with output YES.
$u \neq v$ in $S \Leftrightarrow$ (P_{no}) eventually terminates with output NO, since S is residually finite.

Universal algebras and varieties

Definition

A universal algebra $\mathcal{A}$ is a tuple $(A; f_1, \ldots, f_m, c_1, \ldots, c_k)$, where $A \neq \varnothing$ is the doman of $\mathcal{A}$, and

- Each f_i is a function
$$f_i : A^{n_i} \to A.$$
These are the basic operations of the algebra.
- Each c_j is a constant (formally the image of a nullary function).

The signature of $\mathcal{A}$ is $(f_1, \ldots, f_m, c_1, \ldots, c_k)$. Sometimes an algebraic signature is regarded as simply a list of arities.

Definition

A variety of algebras is the class of all algebraic structures of a given signature satisfying a given set of identities.

Semigroups, monoids and groups as varieties

- Semigroups form a variety of algebras of signature (2) (a single binary operation), defined by the identity

$$x(yz) = (xy)z.$$

- Monoids form a variety of algebras of signature $(2,0)$ the two operations being respectively multiplication (binary), and identity (nullary, a constant), defined by the set of identities

$$\begin{gathered} x(yz) = (xy)z \\ 1x = x1. \end{gathered}$$

- Groups form a variety of algebras of signature $(2,0,1)$ the two operations being respectively multiplication (binary), identity (nullary, a constant), and inversion (unary), defined by the set of identities

$$\begin{gathered} x(yz) = (xy)z \\ 1x = x1 \\ x^{-1}x = xx^{-1} = 1. \end{gathered}$$

- Rings form a variety of algebras with signature $(2,2,0,0,1)$.

Finite presentations and the word problem in general

- $\mathcal{V}$: a variety of universal algebras
- F_X: the finitely generated free object in $\mathcal{V}$ with set X if free generators (it exists, and is defined using the usual universal property).
- ρ: congruence on F_X generated by finitely many pairs $R = \{(u_i, v_i) : i \in I\}$ of elements from F_X.
- The factor algebra F_X/ρ is called finitely presented inside $\mathcal{V}$, and is denoted $\langle X \mid R, \mathcal{V}\rangle$.
- The word problem is decidable in F_X/ρ if there is an algorithm which tells us for any pair of elements (x, y) both from F_X whether $(x, y) \in \rho$.

Conclusion

For any fixed variety of universal algebras the notion of presentation by generators and relations may be defined and investigated, leading to the theories of presentations of groups, monoids, semigroups, rings, inverse semigroups, ...

One-relator presentations

Magnus (1932): The word problem is decidable for one relator groups $\mathrm{Gp}\langle A \mid w = 1\rangle$.

Adjan (1966): The word problem is decidable for one relator monoids of the form $\langle A \mid w = 1\rangle$.

Inverse monoids – variety of algebras of signature $(2, 1, 0)$, operations multiplication, $a \mapsto a^{-1}$, and identity, defined by the identities

$$a(bc) = (ab)c,\ (a^{-1})^{-1} = a,\ aa^{-1}a = a,\ (ab)^{-1} = b^{-1}a^{-1}$$

$$(aa^{-1})(bb^{-1}) = (bb^{-1})(aa^{-1}),\ 1a = a1 = a$$

These are the algebraic model for the study of partial symmetries.

Conjecture (Margolis, Meakin, Stephen (1987))

If $M = \mathrm{Inv}\langle A \mid w = 1\rangle$, then the word problem for M is decidable.

One-relator presentations

Magnus (1932): The word problem is decidable for one relator groups $\mathrm{Gp}\langle A \mid w = 1\rangle$.

Adjan (1966): The word problem is decidable for one relator monoids of the form $\langle A \mid w = 1\rangle$.

Inverse monoids – variety of algebras of signature $(2,1,0)$, operations multiplication, $a \mapsto a^{-1}$, and identity, defined by the identities

$$a(bc) = (ab)c,\ (a^{-1})^{-1} = a,\ aa^{-1}a = a,\ (ab)^{-1} = b^{-1}a^{-1}$$

$$(aa^{-1})(bb^{-1}) = (bb^{-1})(aa^{-1}),\ 1a = a1 = a$$

These are the algebraic model for the study of partial symmetries.

Conjecture (Margolis, Meakin, Stephen (1987))

If $M = \mathrm{Inv}\langle A \mid w = 1\rangle$, then the word problem for M is decidable.

Theorem (RDG (2019))

There is a one-relator inverse monoid $\mathrm{Inv}\langle A \mid w = 1\rangle$ with undecidable word problem.