# Computation, Encoding and Languages

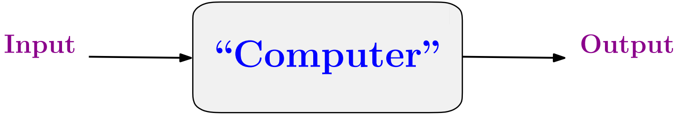
- Computational Problems, Strings and Data Encoding
- Binary Encoding
- Language
- Versions of Computational Problems
- Decision Problems as Language Recognition
- Models of Computation – CPU + Memory

Imdad ullah Khan

# Data Encoding in Computation

# What is computation?

Computation: Processing information by applying a finite set of rules



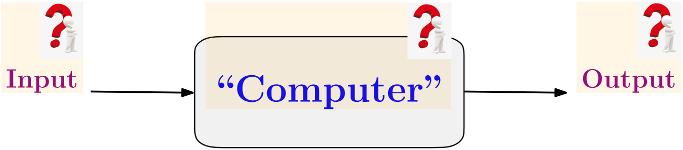

Description of Processing is called Algorithm that converts the input to the desired output

Different set of rules/operations lead to different computational capabilities and limits

Information needs to be encoded to be input for application of rules/operations

# What is computation?

Computation: Processing information by applying a finite set of rules




Description of Processing is called Algorithm that converts the input to the desired output

Different set of rules/operations lead to different computational capabilities and limits

Information needs to be encoded to be input for application of rules/operations

# What is a computational problem?

A computational problem is characterized by three things

- $\mathcal{I}$: set of (valid) input instances
- $S$: solution space, set of possible solutions for instances in $\mathcal{I}$
- $f : \mathcal{I} \mapsto S$: The computational question or function


| Instance $\in \mathcal{I}$ | Solution $\in S$ |
|---|---|
| 0 | No |
| 1 | No |
| 2 | Yes |
| 3 | Yes |
| 4 | No |
| 5 | Yes |
| 6 | No |
| 7 | Yes |
| $\vdots$ | $\vdots$ |


| Instance $\in \mathcal{I}$ | Solution $\in S$ |
|---|---|
| $0, 0$ | 0 |
| $0, 1$ | 0 |
| $1, 2$ | 2 |
| $2, 3$ | 6 |
| $2, 7$ | 14 |
| $2, 5$ | 10 |
| $4, 6$ | 24 |
| $9, 7$ | 63 |
| $\vdots$ | $\vdots$ |

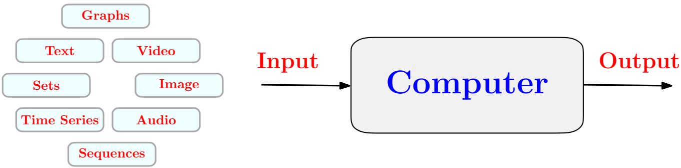

| Instance $\in \mathcal{I}$ | Solution $\in S$ |
|---|---|
| $[0, 3, 2, 9, 5]$ | $[0, 2, 3, 5, 9]$ |
| $[9, 8, 6, 4, 3, 1]$ | $[1, 3, 4, 6, 8, 9]$ |
| $[1, 2, .3, 8, 7.4]$ | $[1, 2, .3, 7.4, 8]$ |
| $[-1, 2, 9, 7, 6]$ | $[-1, 2, 6, 7, 9]$ |
| $[4, 5, 6, 7, 8]$ | $[4, 5, 6, 7, 8]$ |
| $[-2.3, -5.9, -.4.3]$ | $[-5.9, -.4.3, -2.3]$ |
| $[7, -3.6, 9, .8, .5]$ | $[-3.6, .5, .8, 7, 9]$ |
| $\vdots$ | $\vdots$ |

Computation: Processing information by applying a finite set of rules

Description of Processing is called Algorithm that converts the input to the desired output

Different set of rules/operations lead to different computational capabilities and limits

Information needs to be encoded to be input for application of rules/operations




**How do we represent/encode input and output?**

# Data Encoding

## Computation requires data representation

We have already done it for written communication in English

$\Sigma$


string representations

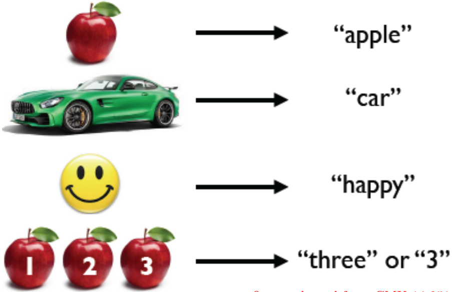



figure adapted from CMU 14-251

What if we had fewer/more symbols?

Can all data be represented as bits ($\Sigma = \{0, 1\}$)?

## String

A **String** is a finite concatenation of symbols from $\Sigma$

Symbol : Examples $a, b, 0, 1, x, /, \#$

Alphabet : A finite set $\Sigma$ of all valid symbols

$\Sigma = \{0, 1\}$

$\Sigma = \{a, b, c\}$

$\Sigma = \{A, B, \ldots Z, a, b, \ldots, z, 0, 2, \ldots, 9, `,', `.', `;', \ldots\}$

$\Sigma^* =$ set of all strings from $\Sigma$

$000, 0, 0101, 110111, 11111, \ldots$

# String

A **String** is a finite concatenation of symbols from $\Sigma$

Length of a string is the number of characters/symbols in it

$|abbba| = 5 \qquad |01| = 2, \qquad |0101abc\#\$abc| = 12$

Empty String ($\epsilon$) is a string with no symbols

$|\epsilon| = 0$

## String

A **String** is a finite concatenation of symbols from $\Sigma$

Concatenation of $x, y \in \Sigma^*$ (denoted by $xy$) is $x$ followed by $y$

$x = aaaaabbbabab \qquad y = bababa \qquad xy = aaaaabbbabab bababa$

$x = abc \quad y = \epsilon \quad xy = abc$

Reversal of $x \in \Sigma^*$ ($x^R$) consists of symbols of $x$ written backwards

$x = aaabb \quad x^R = bbaaa \qquad x = \epsilon \quad x^R = \epsilon$

What is $(xy)^R$ ?

# Data Encoding

An encoding/representation scheme for a set of objects $O$ is a one-to-one function $E : O \mapsto \{0,1\}^*$

Encoding should be one-to-one for decoding

$$D : range(E) \mapsto O \quad s.t \quad D(E(x)) = x \quad \forall x \in O$$

Does every object have a corresponding encoding ?

Can two objects have the same encoding ?

Does every string correspond to a valid encoding ?

Does $\Sigma$ make a difference ?

Does $|\Sigma|$ make a difference ?

# Data Encoding

An encoding/representation scheme for a set of objects $O$ is a one-to-one function $E : O \mapsto \{0,1\}^*$

Encoding should be one-to-one for decoding

$$D : range(E) \mapsto O \quad s.t \quad D(E(x)) = x \quad \forall x \in O$$

"Good" representation scheme is active research area in coding theory, information theory (theoretical) and representation learning (practical)

- Compression: Representation with small size (e.g., JPEG)
- Error correction: Representation that is robust to errors (e.g., "control digits", error correcting codes)
- Efficiency: Representation enabling fast operations (e.g., binary numbers, distance oracles)
- Feature extraction: Representation enabling data analytics
- Secrecy: Representation hiding certain information (e.g., encryption)