Université Paris Cité – LIPADE

Algorithmic Complexity

Introduction

Jean-Guy Mailly (jean-guy.mailly@u-paris.fr)

2022

Infos sur l'UE

- Jean-Guy Mailly : jean-guy.mailly@u-paris.fr
  Bureau 814 I
- 15h de cours : lundi, 10h15–11h45, Curie A
- 15h de TD : lundi, 12h45–14h15, Cuneo D (RSA/DCI)
  vendredi, 10h30–12h00, Vieussens D (IAD/VMI)
- Modalités de contrôle de connaissances :
  - Un contrôle continu vers la mi-semestre
  - Un examen en fin de semestre
  - Note finale : $\max(EX, \frac{EX+CC}{2})$
- Moodle: Cours IFFAX020 Complexité Algorithmique

  `https://moodle.u-paris.fr/course/view.php?id=5486`

Aims

Algorithmic Complexity

Measure the hardness of a problem w.r.t. the efficiency of algorithms to solve it

- Time
- Space

Goal of the Course

- Bases of complexity theory
- Main complexity classes (time and space)
- Complexity of usual problems
- Being able to determine the complexity of a problem

Resources

When solving a problem, two kinds of resources are used

- time
  - number of seconds
  - number of steps for the computation
- space
  - number of bytes used to execute the program
  - number of variables used to represent and solve the problem

Complexity Theory

- Classification of problems w.r.t. the resources required to solve them
  - The more we need time and/or space, the harder it is
- Comparison of problems (depending on the class they belong)
- Solving problems by translating them into other problems (with the same complexity)

Outline

Mappings and Asymptotic Bounds

Integer mappings

We use mappings $f : \mathbb{N} \rightarrow \mathbb{N}$ to represent the time (or space) used to solve a problem

- Intuition: If the problem entry has size n, $f(n)$ steps are required to compute the result
- If needed, we use the closest integer value (e.g. $\log(n)$ means $\lceil \log(n) \rceil$)

Mappings and Asymptotic Bounds

Integer mappings

We use mappings $f : \mathbb{N} \to \mathbb{N}$ to represent the time (or space) used to solve a problem

- Intuition: If the problem entry has size n, $f(n)$ steps are required to compute the result
- If needed, we use the closest integer value (e.g. $\log(n)$ means $\lceil \log(n) \rceil$)

Asymptotic bounds – $\mathcal{O}$ notation

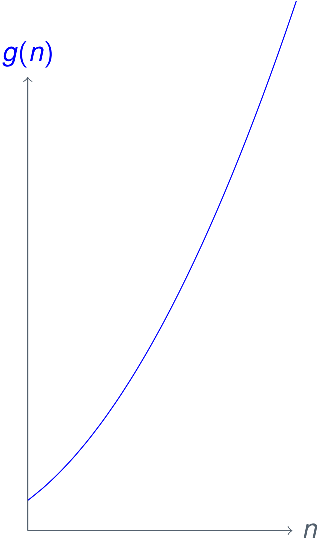
Given a mapping f, $\mathcal{O}(f(n))$ is the set of mappings g s.t. $\exists n_o, c$,

$$\forall n \geq n_0, g(n) \leq c \times f(n)$$

- Intuition: When n is large enough, g is smaller than f modulo some constant c

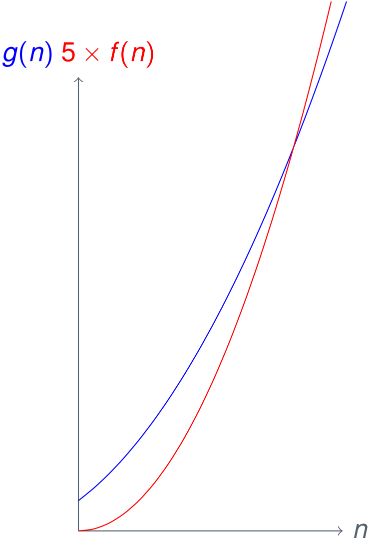
Example

- We suppose that we have an algorithm which solves a graphs problem in $g(n) = 4 \times n^2 + 3 \times n + 2$ steps, when n is the size of the graph (i.e. number of vertices)



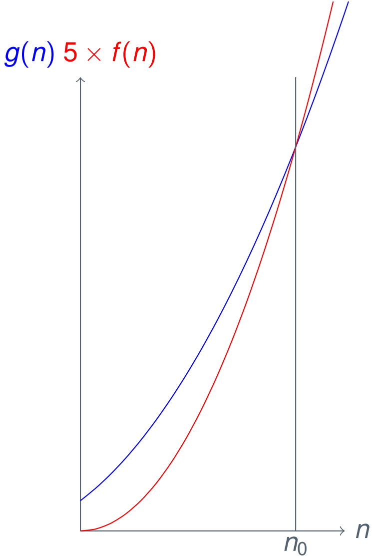
Example

- We suppose that we have an algorithm which solves a graphs problem in $g(n) = 4 \times n^2 + 3 \times n + 2$ steps, when n is the size of the graph (i.e. number of vertices)
- When n becomes large enough, $g(n) \leq 5 \times f(n) = 5 \times n^2$, so $g(n) \in \mathcal{O}(n^2)$



Example

- We suppose that we have an algorithm which solves a graphs problem in $g(n) = 4 \times n^2 + 3 \times n + 2$ steps, when n is the size of the graph (i.e. number of vertices)
- When n becomes large enough, $g(n) \leq 5 \times f(n) = 5 \times n^2$, so $g(n) \in \mathcal{O}(n^2)$



On Usual $\mathcal{O}$ Families

From Slow to Fast Increase

C is an arbitrary constant, and $\log$ is any logarithmic function

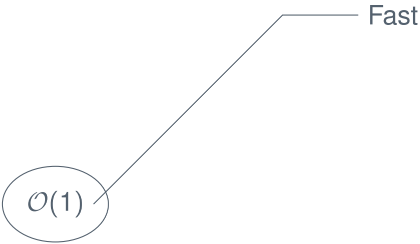
Family of Functions	Name
$\mathcal{O}(1)$	Constant
$\mathcal{O}(\log(n))$	Logarithmic
$\mathcal{O}((\log(n))^c)$	Polylogarithmic
$\mathcal{O}(n)$	Linear
$\mathcal{O}(n\log(n))$	Linearithmic
$\mathcal{O}(n^2)$	Quadratic
$\mathcal{O}(n^C)$	Polynomial
$\mathcal{O}(C^n)$	Exponential
$\mathcal{O}(n!)$	Factorial

On the Sum of Functions

If $f(n) = g(n) + h(n)$, $\mathcal{O}(f(n)) = \max(\mathcal{O}(g(n), \mathcal{O}(h(n)))$

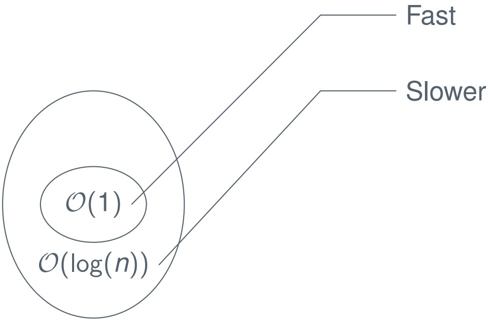
Explanation of the $\mathcal{O}$ "Hierarchy"

- $\mathcal{O}(1) = \{g \mid \exists n_0, c, \forall n \geq n_0, g(n) \leq c \times 1\}$



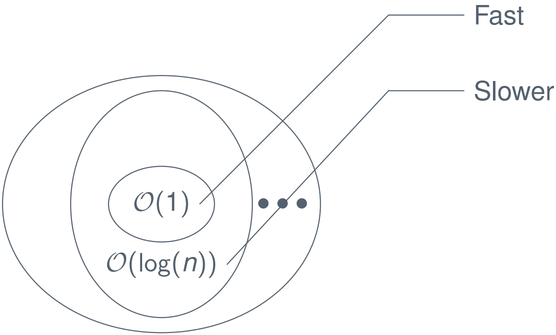
Explanation of the $\mathcal{O}$ "Hierarchy"

- $\mathcal{O}(1) = \{g \mid \exists n_0, c, \forall n \geq n_0, g(n) \leq c \times 1\}$
- $\mathcal{O}(\log(n)) = \{g \mid \exists n_0, c, \forall n \geq n_0, g(n) \leq c \times \log(n)\}$



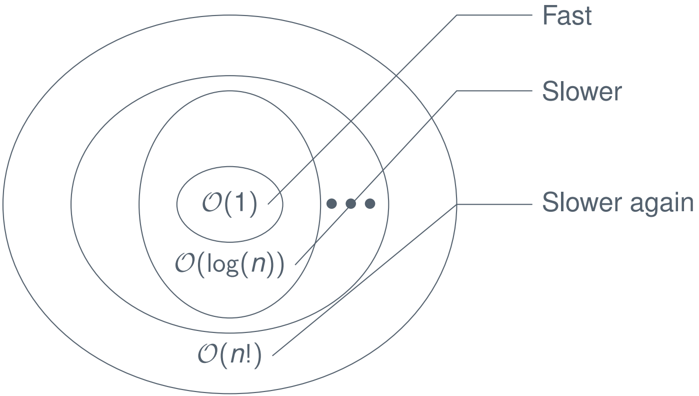
Explanation of the $\mathcal{O}$ "Hierarchy"

- $\mathcal{O}(1) = \{g \mid \exists n_0, c, \forall n \geq n_0, g(n) \leq c \times 1\}$
- $\mathcal{O}(\log(n)) = \{g \mid \exists n_0, c, \forall n \geq n_0, g(n) \leq c \times \log(n)\}$
- $\ldots$



Explanation of the $\mathcal{O}$ "Hierarchy"

- $\mathcal{O}(1) = \{g \mid \exists n_0, c, \forall n \geq n_0, g(n) \leq c \times 1\}$
- $\mathcal{O}(\log(n)) = \{g \mid \exists n_0, c, \forall n \geq n_0, g(n) \leq c \times \log(n)\}$
- $\ldots$
- $\mathcal{O}(n!) = \{g \mid \exists n_0, c, \forall n \geq n_0, g(n) \leq c \times n!\}$



Outline

Different Kinds of Problems (1/2)

Decision Problem, Function Problem

A *decision problem* over a set of input data E is a mapping from any element in E to a value in $\{false, true\}$ ($\simeq \{0, 1\} \simeq$ {NO,YES}).
A *function problem* over a set of input data E is a mapping from any element in E to a single outcome.

Solving an Equation

Decision Problem Does the equation $f(x) = y$ have a solution?

Function Problem Give a solution of the equation $f(x) = y$

Different Kinds of Problems (2/2)

Enumeration Problem, Optimization Problem

Given a function problem $\mathcal{P}$,
The *enumeration problem* ENUM-$\mathcal{P}$ over E is a mapping from any element $e_i \in E$ to the set of all outcome of $\mathcal{P}$ over e_i.
The *optimization problem* OPT-$\mathcal{P}$ over E is a mapping from any element in E to a single outcome which minimizes a given criterion.

Solving an Equation

Enumeration Problem Give all the solutions of the equation $f(x) = y$

Optimization Problem Give a minimal solution of the equation $f(x) = y$

Languages

Definition

- Set of symbols Σ called *vocabulary* or *alphabet*.
- A *word* w is a sequence of symbols $w_1 w_2 \dots w_k$, with $w_i \in \Sigma$ for all i.
- $\Sigma^* = \{w_1 w_2 \dots w_k \mid w_i \in \Sigma, k \in \mathbb{N}\}$
- a *language* $\mathcal{L}$ is any subset of Σ^*, the *complement* of $\mathcal{L}$ is $\bar{\mathcal{L}} = \Sigma^* \setminus \mathcal{L}$

Language $\simeq$ Decision Problem

- For any language $\mathcal{L}$, $\mathcal{P}(\mathcal{L})$ is the decision problem:

  Given $x \in \Sigma^*$, does x belong to $\mathcal{L}$?
- For any decision problem $\mathcal{P}$, $\mathcal{L}(\mathcal{P})$ is the language:

  $\{x \in \text{instances of } \mathcal{P} \mid x \text{ is a positive instance of } \mathcal{P}\}$

Outline

Non-Directed Graphs

Definition

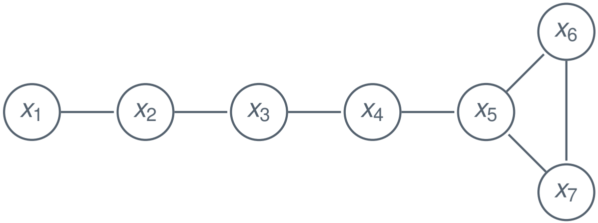
A non-directed graph is a pair $G = \langle N, E \rangle$ where N is the set of nodes and $E \subseteq pairs(N)$ is the set of edges, with $pairs(N) = \{\{x_i, x_j\} \mid x_i, x_j \in N\}$

Non-Directed Graphs

Definition

A non-directed graph is a pair $G = \langle N, E \rangle$ where N is the set of nodes and $E \subseteq pairs(N)$ is the set of edges, with
$pairs(N) = \{\{x_i, x_j\} \mid x_i, x_j \in N\}$

$G = \langle N, E \rangle$, with $N = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$ and
$E = \{\{x_1, x_2\}, \{x_2, x_3\}, \{x_3, x_4\}, \{x_4, x_5\}, \{x_5, x_6\}, \{x_6, x_7\}, \{x_7, x_5\}\}$



Subgraphs

Definition

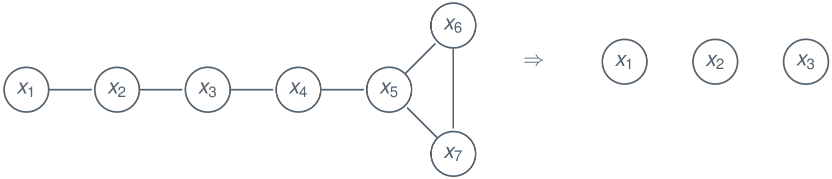
A subgraph in a non-directed graph $G = \langle N, E \rangle$ is a pair $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq pairs(N') \cap E$. If $E' = pairs(N') \cap E$, we say that G' is the subgraph of G induced by N'

Subgraphs

Definition

A subgraph in a non-directed graph $G = \langle N, E \rangle$ is a pair $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq pairs(N') \cap E$. If $E' = pairs(N') \cap E$, we say that G' is the subgraph of G induced by N'

$G' = \langle \{x_1, x_2, x_3\}, \{\{x_1, x_2\}\} \rangle$ is a subgraph of G

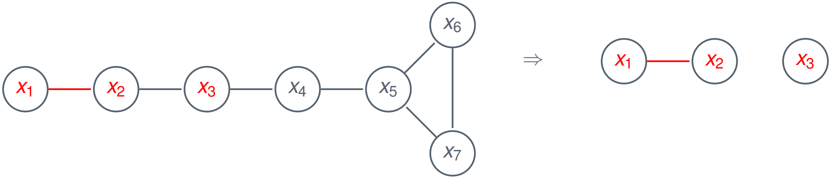


Subgraphs

Definition

A subgraph in a non-directed graph $G = \langle N, E\rangle$ is a pair $G' = \langle N', E'\rangle$, with $N' \subseteq N$ and $E' \subseteq pairs(N') \cap E$. If $E' = pairs(N') \cap E$, we say that G' is the subgraph of G induced by N'

$G' = \langle\{x_1, x_2, x_3\}, \{\{x_1, x_2\}\}\rangle$ is a subgraph of G

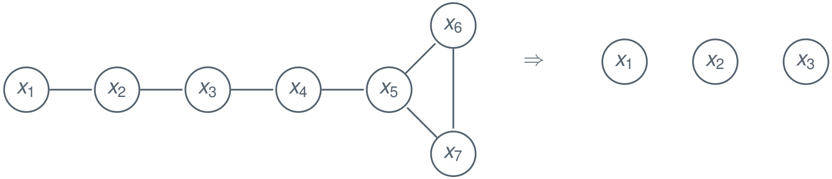


Subgraphs

Definition

A subgraph in a non-directed graph $G = \langle N, E \rangle$ is a pair $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq pairs(N') \cap E$. If $E' = pairs(N') \cap E$, we say that G' is the subgraph of G induced by N'

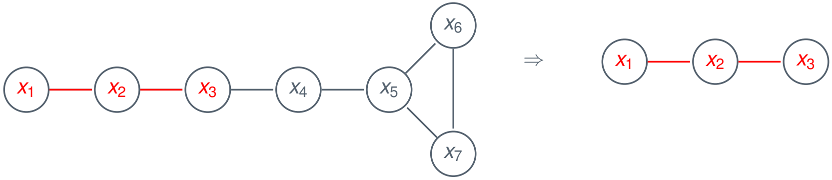
$G' = \langle \{x_1, x_2, x_3\}, \{\{x_1, x_2\}\} \rangle$ is a subgraph of G
$G' = \langle \{x_1, x_2, x_3\}, \{\{x_1, x_2\}, \{x_2, x_3\}\} \rangle$ is the subgraph of G induced by $\{x_1, x_2, x_3\}$



Definition

A subgraph in a non-directed graph $G = \langle N, E \rangle$ is a pair $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq pairs(N') \cap E$. If $E' = pairs(N') \cap E$, we say that G' is the subgraph of G induced by N'

$G' = \langle \{x_1, x_2, x_3\}, \{\{x_1, x_2\}\} \rangle$ is a subgraph of G
$G' = \langle \{x_1, x_2, x_3\}, \{\{x_1, x_2\}, \{x_2, x_3\}\} \rangle$ is the subgraph of G induced by $\{x_1, x_2, x_3\}$



Chain

Definition

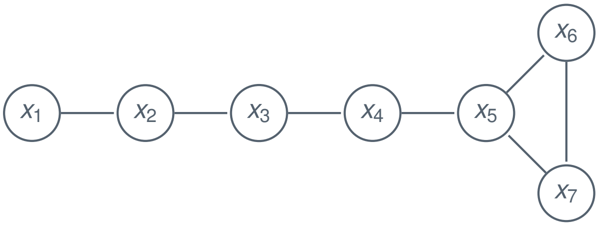
A chain in a non-directed graph $G = \langle N, E \rangle$ is a vector of nodes $(n_1, \ldots, n_k)$ such that $\forall 0 \leq i < k$, $\{n_i, n_{i+1}\} \in E$.

Chain

Definition

A chain in a non-directed graph $G = \langle N, E \rangle$ is a vector of nodes $(n_1, \ldots, n_k)$ such that $\forall 0 \leq i < k$, $\{n_i, n_{i+1}\} \in E$.

(x_3, x_4, x_5, x_6) and (x_3, x_4, x_5, x_7) are chains of G

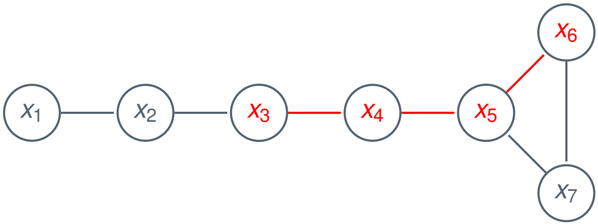


Chain

Definition

A chain in a non-directed graph $G = \langle N, E \rangle$ is a vector of nodes $(n_1, \ldots, n_k)$ such that $\forall 0 \leq i < k$, $\{n_i, n_{i+1}\} \in E$.

(x_3, x_4, x_5, x_6) and (x_3, x_4, x_5, x_7) are chains of G

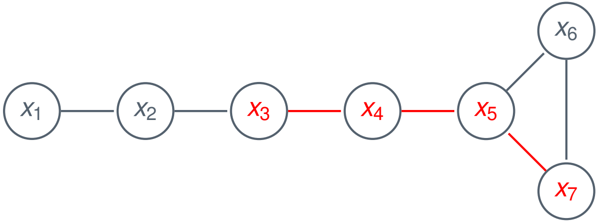


Chain

Definition

A chain in a non-directed graph $G = \langle N, E \rangle$ is a vector of nodes $(n_1, \ldots, n_k)$ such that $\forall 0 \leq i < k$, $\{n_i, n_{i+1}\} \in E$.

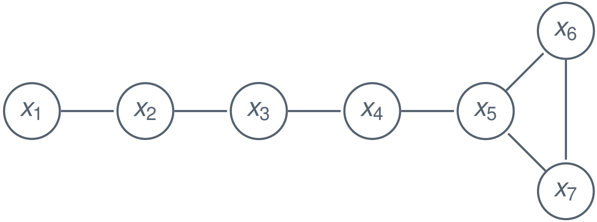
(x_3, x_4, x_5, x_6) and (x_3, x_4, x_5, x_7) are chains of G



Definition

A chain in a non-directed graph $G = \langle N, E \rangle$ is a vector of nodes $(n_1, \ldots, n_k)$ such that $\forall 0 \leq i < k$, $\{n_i, n_{i+1}\} \in E$.

(x_3, x_4, x_5, x_6) and (x_3, x_4, x_5, x_7) are chains of G
(x_1, x_6) is not a chain of G



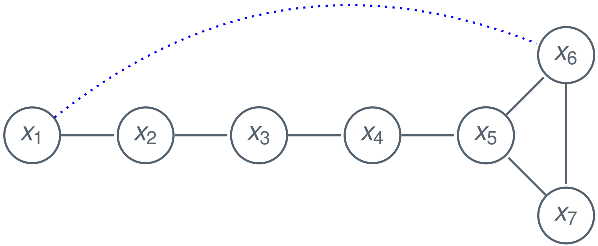
Chain

Definition

A chain in a non-directed graph $G = \langle N, E \rangle$ is a vector of nodes $(n_1, \ldots, n_k)$ such that $\forall 0 \leq i < k$, $\{n_i, n_{i+1}\} \in E$.

(x_3, x_4, x_5, x_6) and (x_3, x_4, x_5, x_7) are chains of G

(x_1, x_6) is not a chain of G



Clique

Definition

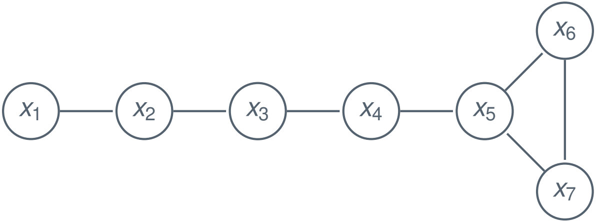
A clique in a non-directed graph $G = \langle N, E \rangle$ is subgraph $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq pairs(N') \cap E$, such that $\forall x_i, x_j \in N', x_i \neq x_j, \{x_i, x_j\} \in E'$

Clique

Definition

A clique in a non-directed graph $G = \langle N, E\rangle$ is subgraph $G' = \langle N', E'\rangle$, with $N' \subseteq N$ and $E' \subseteq pairs(N') \cap E$, such that $\forall x_i, x_j \in N', x_i \neq x_j, \{x_i, x_j\} \in E'$

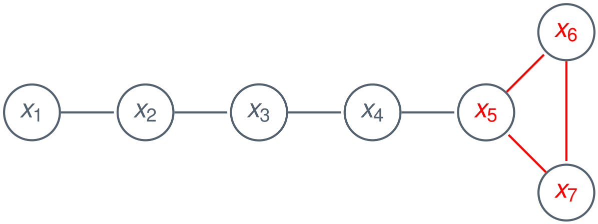
$\langle\{x_5, x_6, x_7\}, \{\{x_5, x_6\}, \{x_6, x_7\}, \{x_7, x_5\}, \}\rangle$ is a clique in G



Definition

A clique in a non-directed graph $G = \langle N, E \rangle$ is subgraph $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq pairs(N') \cap E$, such that $\forall x_i, x_j \in N', x_i \neq x_j, \{x_i, x_j\} \in E'$

$\langle \{x_5, x_6, x_7\}, \{\{x_5, x_6\}, \{x_6, x_7\}, \{x_7, x_5\}, \} \rangle$ is a clique in G



Cycle

Definition

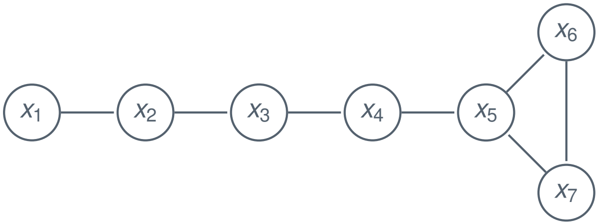
A cycle in a non-directed graph $G = \langle N, E \rangle$ is a chain $(n_1, \ldots, n_k)$ such that $n_1 = n_k$. The length of the cycle is $k - 1$

Cycle

Definition

A cycle in a non-directed graph $G = \langle N, E \rangle$ is a chain $(n_1, \ldots, n_k)$ such that $n_1 = n_k$. The length of the cycle is $k - 1$

(x_5, x_6, x_7, x_5) is a cycle of length 3 in G

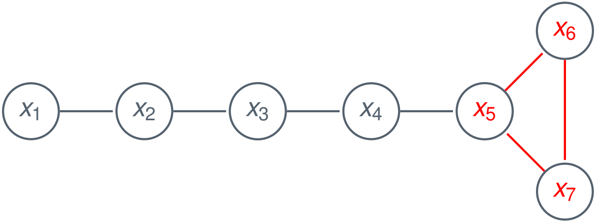


Cycle

Definition

A cycle in a non-directed graph $G = \langle N, E \rangle$ is a chain $(n_1, \ldots, n_k)$ such that $n_1 = n_k$. The length of the cycle is $k - 1$

(x_5, x_6, x_7, x_5) is a cycle of length 3 in G



Directed Graphs

Definition

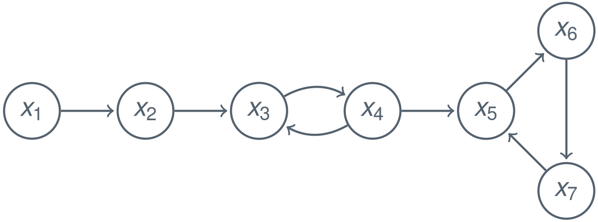
A directed graph is a pair $G = \langle N, E \rangle$ where N is the set of nodes and $E \subseteq N \times N$ is the set of edges

Directed Graphs

Definition

A directed graph is a pair $G = \langle N, E \rangle$ where N is the set of nodes and $E \subseteq N \times N$ is the set of edges

$G = \langle N, E \rangle$, with $N = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$ and $E = \{(x_1, x_2), (x_2, x_3), (x_3, x_4), (x_4, x_3), (x_4, x_5), (x_5, x_6), (x_6, x_7), (x_7, x_5)\}$



Subgraphs

Definition

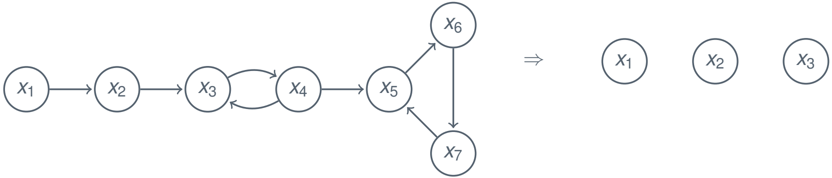
A subgraph in a directed graph $G = \langle N, E \rangle$ is a pair $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq (N' \times N') \cap E$. If $E' = (N' \times N') \cap E$, we say that G' is the subgraph of G induced by N'

Subgraphs

Definition

A subgraph in a directed graph $G = \langle N, E \rangle$ is a pair $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq (N' \times N') \cap E$. If $E' = (N' \times N') \cap E$, we say that G' is the subgraph of G induced by N'

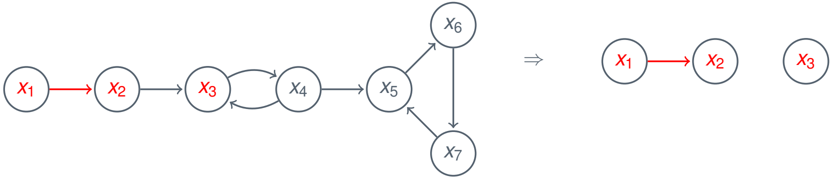
$G' = \langle \{x_1, x_2, x_3\}, \{(x_1, x_2)\} \rangle$ is a subgraph of G



Definition

A subgraph in a directed graph $G = \langle N, E\rangle$ is a pair $G' = \langle N', E'\rangle$, with $N' \subseteq N$ and $E' \subseteq (N' \times N') \cap E$. If $E' = (N' \times N') \cap E$, we say that G' is the subgraph of G induced by N'

$G' = \langle\{x_1, x_2, x_3\}, \{(x_1, x_2)\}\rangle$ is a subgraph of G

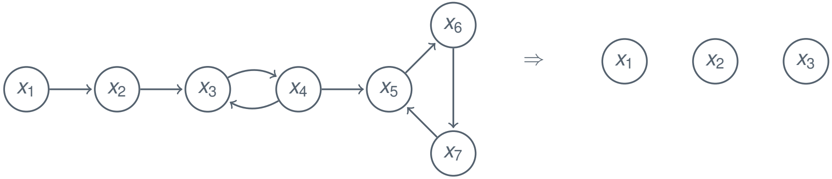


Subgraphs

Definition

A subgraph in a directed graph $G = \langle N, E \rangle$ is a pair $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq (N' \times N') \cap E$. If $E' = (N' \times N') \cap E$, we say that G' is the subgraph of G induced by N'

$G' = \langle \{x_1, x_2, x_3\}, \{(x_1, x_2)\} \rangle$ is a subgraph of G
$G' = \langle \{x_1, x_2, x_3\}, \{(x_1, x_2), (x_2, x_3)\} \rangle$ is the subgraph of G induced by $\{x_1, x_2, x_3\}$

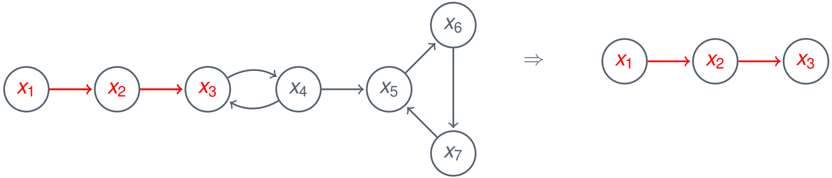


Subgraphs

Definition

A subgraph in a directed graph $G = \langle N, E \rangle$ is a pair $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq (N' \times N') \cap E$. If $E' = (N' \times N') \cap E$, we say that G' is the subgraph of G induced by N'

$G' = \langle \{x_1, x_2, x_3\}, \{(x_1, x_2)\} \rangle$ is a subgraph of G
$G' = \langle \{x_1, x_2, x_3\}, \{(x_1, x_2), (x_2, x_3)\} \rangle$ is the subgraph of G induced by $\{x_1, x_2, x_3\}$



Path

Definition

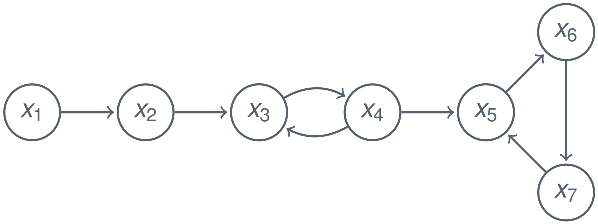
A path in a directed graph $G = \langle N, E \rangle$ is a vector of nodes $(n_1, \ldots, n_k)$ such that $\forall 0 \leq i < k$, $(n_i, n_{i+1}) \in E$.

Path

Definition

A path in a directed graph $G = \langle N, E \rangle$ is a vector of nodes $(n_1, \ldots, n_k)$ such that $\forall 0 \leq i < k$, $(n_i, n_{i+1}) \in E$.

(x_3, x_4, x_5, x_6) is a path of G, but (x_3, x_4, x_5, x_7) is not

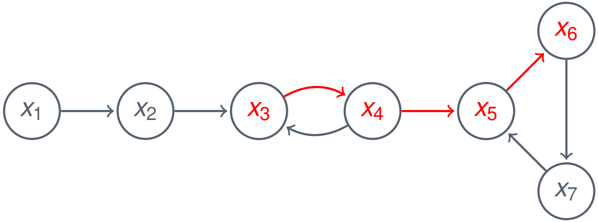


Path

Definition

A path in a directed graph $G = \langle N, E \rangle$ is a vector of nodes $(n_1, \ldots, n_k)$ such that $\forall 0 \leq i < k$, $(n_i, n_{i+1}) \in E$.

(x_3, x_4, x_5, x_6) is a path of G, but (x_3, x_4, x_5, x_7) is not

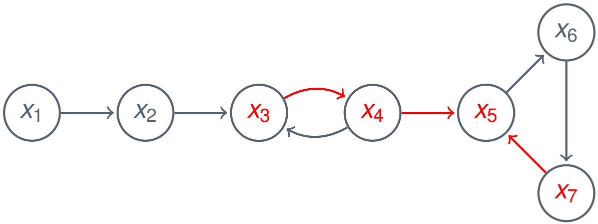


Path

Definition

A path in a directed graph $G = \langle N, E \rangle$ is a vector of nodes $(n_1, \ldots, n_k)$ such that $\forall 0 \leq i < k$, $(n_i, n_{i+1}) \in E$.

(x_3, x_4, x_5, x_6) is a path of G, but (x_3, x_4, x_5, x_7) is not

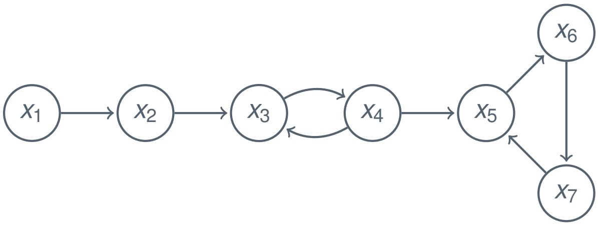


Path

Definition

A path in a directed graph $G = \langle N, E \rangle$ is a vector of nodes $(n_1, \dots, n_k)$ such that $\forall 0 \leq i < k$, $(n_i, n_{i+1}) \in E$.

(x_3, x_4, x_5, x_6) is a path of G, but (x_3, x_4, x_5, x_7) is not
(x_1, x_6) is not a path of G

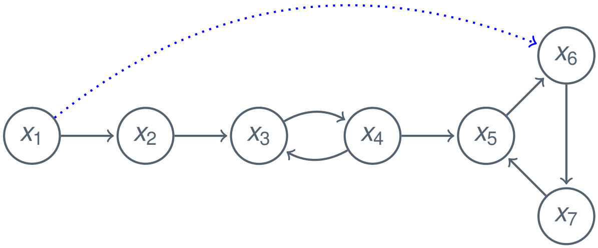


Path

Definition

A path in a directed graph $G = \langle N, E \rangle$ is a vector of nodes $(n_1, \dots, n_k)$ such that $\forall 0 \leq i < k$, $(n_i, n_{i+1}) \in E$.

(x_3, x_4, x_5, x_6) is a path of G, but (x_3, x_4, x_5, x_7) is not
(x_1, x_6) is not a path of G



Clique

Definition

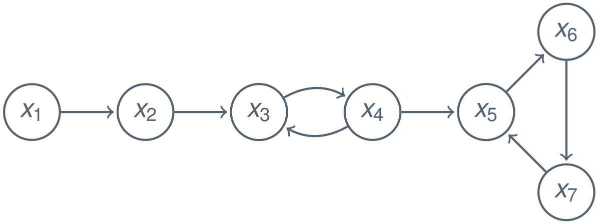
A clique in a directed graph $G = \langle N, E \rangle$ is subgraph $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq (N' \times N') \cap E$, such that $\forall x_i, x_j \in N', x_i \neq x_j$, $(x_i, x_j) \in E'$

Clique

Definition

A clique in a directed graph $G = \langle N, E\rangle$ is subgraph $G' = \langle N', E'\rangle$, with $N' \subseteq N$ and $E' \subseteq (N' \times N') \cap E$, such that $\forall x_i, x_j \in N', x_i \neq x_j$, $(x_i, x_j) \in E'$

$\langle\{x_5, x_6, x_7\}, \{(x_5, x_6), (x_6, x_7), (x_7, x_5)\}\rangle$ is not a clique in G

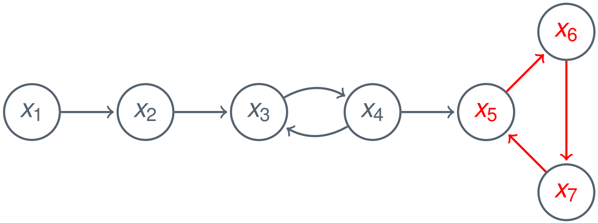


Clique

Definition

A clique in a directed graph $G = \langle N, E \rangle$ is subgraph $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq (N' \times N') \cap E$, such that $\forall x_i, x_j \in N', x_i \neq x_j$, $(x_i, x_j) \in E'$

$\langle \{x_5, x_6, x_7\}, \{(x_5, x_6), (x_6, x_7), (x_7, x_5)\} \rangle$ is not a clique in G

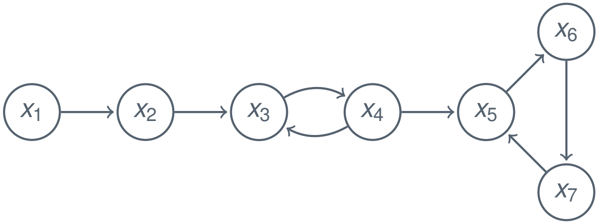


Clique

Definition

A clique in a directed graph $G = \langle N, E \rangle$ is subgraph $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq (N' \times N') \cap E$, such that $\forall x_i, x_j \in N', x_i \neq x_j$, $(x_i, x_j) \in E'$

$\langle \{x_5, x_6, x_7\}, \{(x_5, x_6), (x_6, x_7), (x_7, x_5)\} \rangle$ is not a clique in G
$\langle \{x_3, x_4\}, \{(x_3, x_4), (x_4, x_3)\} \rangle$ is a clique in G

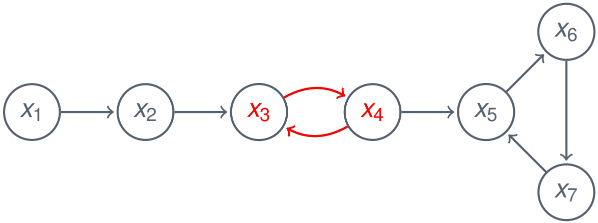


Clique

Definition

A clique in a directed graph $G = \langle N, E \rangle$ is subgraph $G' = \langle N', E' \rangle$, with $N' \subseteq N$ and $E' \subseteq (N' \times N') \cap E$, such that $\forall x_i, x_j \in N', x_i \neq x_j$, $(x_i, x_j) \in E'$

$\langle \{x_5, x_6, x_7\}, \{(x_5, x_6), (x_6, x_7), (x_7, x_5)\} \rangle$ is not a clique in G
$\langle \{x_3, x_4\}, \{(x_3, x_4), (x_4, x_3)\} \rangle$ is a clique in G



Cycle

Definition

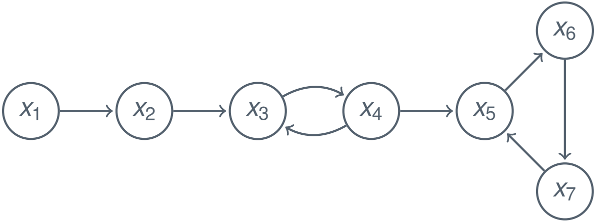
A cycle in a directed graph $G = \langle N, E \rangle$ is a path $(n_1, \ldots, n_k)$ such that $n_1 = n_k$. The length of the cycle is $k - 1$

Cycle

Definition

A cycle in a directed graph $G = \langle N, E \rangle$ is a path $(n_1, \ldots, n_k)$ such that $n_1 = n_k$. The length of the cycle is $k - 1$

(x_3, x_4, x_3) is a cycle of length 2 in G
(x_5, x_6, x_7, x_5) is a cycle of length 3 in G

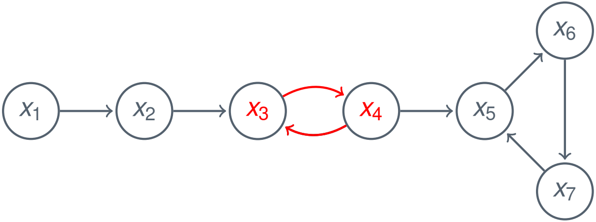


Cycle

Definition

A cycle in a directed graph $G = \langle N, E \rangle$ is a path $(n_1, \ldots, n_k)$ such that $n_1 = n_k$. The length of the cycle is $k - 1$

(x_3, x_4, x_3) is a cycle of length 2 in G
(x_5, x_6, x_7, x_5) is a cycle of length 3 in G

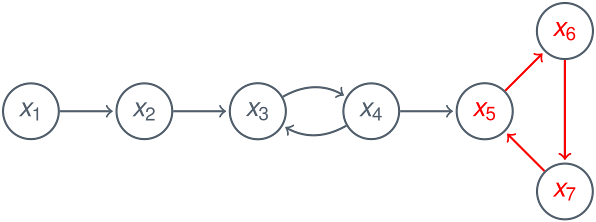


Cycle

Definition

A cycle in a directed graph $G = \langle N, E \rangle$ is a path $(n_1, \ldots, n_k)$ such that $n_1 = n_k$. The length of the cycle is $k - 1$

(x_3, x_4, x_3) is a cycle of length 2 in G
(x_5, x_6, x_7, x_5) is a cycle of length 3 in G

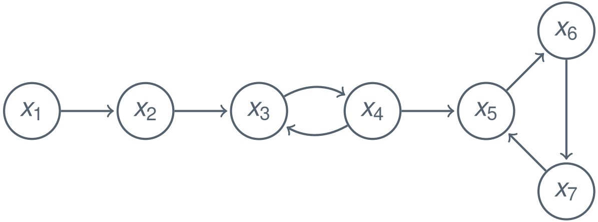


Cycle

Definition

A cycle in a directed graph $G = \langle N, E \rangle$ is a path $(n_1, \ldots, n_k)$ such that $n_1 = n_k$. The length of the cycle is $k - 1$

(x_3, x_4, x_3) is a cycle of length 2 in G
(x_5, x_6, x_7, x_5) is a cycle of length 3 in G
(x_2, x_3, x_4, x_3) is not a cycle of G

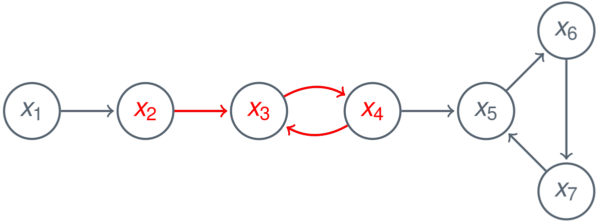


Cycle

Definition

A cycle in a directed graph $G = \langle N, E \rangle$ is a path $(n_1, \ldots, n_k)$ such that $n_1 = n_k$. The length of the cycle is $k - 1$

(x_3, x_4, x_3) is a cycle of length 2 in G
(x_5, x_6, x_7, x_5) is a cycle of length 3 in G
(x_2, x_3, x_4, x_3) is not a cycle of G



Outline

Propositional Logic

Definition

Propositional logic is a mathematical formalism which studies **propositions**, i.e. logical objects which can be **true** or **false**. A propositional formula is the combinations of propositions with **connectives**

Propositional Logic

Definition

Propositional logic is a mathematical formalism which studies **propositions**, i.e. logical objects which can be **true** or **false**. A propositional formula is the combinations of propositions with **connectives**

Examples of propositions:

- it's raining
- it's sunny
- I need an umbrella
- I need sunscreen

Propositional Logic

Definition

Propositional logic is a mathematical formalism which studies **propositions**, i.e. logical objects which can be **true** or **false**. A propositional formula is the combinations of propositions with **connectives**

Examples of propositions:

- it's raining
- it's sunny
- I need an umbrella
- I need sunscreen

Propositions are usually represented by *Boolean variables*, i.e. mathematical objects which can be assigned a value from a binary set $\mathbb{B} = \{0, 1\}$ (sometimes written $\{\mathit{False}, \mathit{True}\}$ or $\{\bot, \top\}$)

Syntax of Propositional Logic

Let $V = \{x_1, \ldots, x_n\}$ be a set of Boolean variables

- **atom** $\forall x_i \in V$, x_i is a (well-formed) formula
- **negation** if φ is a formula, then $\neg\varphi$ is a formula
- **conjunction** if φ_1 and φ_2 are formulas, then $\varphi_1 \wedge \varphi_2$ is a formula

Syntax of Propositional Logic

Let $V = \{x_1, \ldots, x_n\}$ be a set of Boolean variables

- **atom** $\forall x_i \in V$, x_i is a (well-formed) formula
- **negation** if φ is a formula, then $\neg\varphi$ is a formula
- **conjunction** if φ_1 and φ_2 are formulas, then $\varphi_1 \wedge \varphi_2$ is a formula
- **disjunction** if φ_1 and φ_2 are formulas, then $\varphi_1 \vee \varphi_2$ is a formula

Syntax of Propositional Logic

Let $V = \{x_1, \ldots, x_n\}$ be a set of Boolean variables

- **atom** $\forall x_i \in V$, x_i is a (well-formed) formula
- **negation** if φ is a formula, then $\neg\varphi$ is a formula
- **conjunction** if φ_1 and φ_2 are formulas, then $\varphi_1 \wedge \varphi_2$ is a formula
- **disjunction** if φ_1 and φ_2 are formulas, then $\varphi_1 \vee \varphi_2$ is a formula
- **implication** if φ_1 and φ_2 are formulas, then $\varphi_1 \Rightarrow \varphi_2$ is a formula

Syntax of Propositional Logic

Let $V = \{x_1, \ldots, x_n\}$ be a set of Boolean variables

- **atom** $\forall x_i \in V$, x_i is a (well-formed) formula
- **negation** if φ is a formula, then $\neg\varphi$ is a formula
- **conjunction** if φ_1 and φ_2 are formulas, then $\varphi_1 \wedge \varphi_2$ is a formula
- **disjunction** if φ_1 and φ_2 are formulas, then $\varphi_1 \vee \varphi_2$ is a formula
- **implication** if φ_1 and φ_2 are formulas, then $\varphi_1 \Rightarrow \varphi_2$ is a formula
- **equivalence** if φ_1 and φ_2 are formulas, then $\varphi_1 \Leftrightarrow \varphi_2$ is a formula

Syntax of Propositional Logic

Let $V = \{x_1, \ldots, x_n\}$ be a set of Boolean variables

- **atom** $\forall x_i \in V$, x_i is a (well-formed) formula
- **negation** if φ is a formula, then $\neg\varphi$ is a formula
- **conjunction** if φ_1 and φ_2 are formulas, then $\varphi_1 \land \varphi_2$ is a formula
- **disjunction** if φ_1 and φ_2 are formulas, then $\varphi_1 \lor \varphi_2$ is a formula
- **implication** if φ_1 and φ_2 are formulas, then $\varphi_1 \Rightarrow \varphi_2$ is a formula
- **equivalence** if φ_1 and φ_2 are formulas, then $\varphi_1 \Leftrightarrow \varphi_2$ is a formula

We use parentheses to simplify the reading and clarify priorities

Syntax of Propositional Logic

Let $V = \{x_1, \ldots, x_n\}$ be a set of Boolean variables

- **atom** $\forall x_i \in V$, x_i is a (well-formed) formula
- **negation** if φ is a formula, then $\neg\varphi$ is a formula
- **conjunction** if φ_1 and φ_2 are formulas, then $\varphi_1 \wedge \varphi_2$ is a formula
- **disjunction** if φ_1 and φ_2 are formulas, then $\varphi_1 \vee \varphi_2$ is a formula
- **implication** if φ_1 and φ_2 are formulas, then $\varphi_1 \Rightarrow \varphi_2$ is a formula
- **equivalence** if φ_1 and φ_2 are formulas, then $\varphi_1 \Leftrightarrow \varphi_2$ is a formula

We use parentheses to simplify the reading and clarify priorities

Formulas

- $x_1, \neg x_1, \neg\neg x_1$
- $x_1 \wedge x_2, x_1 \vee x_2$
- $x_1 \wedge (x_3 \vee (x_4 \Rightarrow x_1))$
- $\varphi \wedge \psi$

Not Formulas

- $x_1\neg, \neg x_1\neg$
- $x_1\wedge, \vee x_1$
- $x_1 \Rightarrow (\vee x_2)$
- $\wedge\psi$

Semantics of Propositional Logic

Let $V = \{x_1, \ldots, x_n\}$ be a set of Boolean variables. An interpretation ω is a mapping from V to $\mathbb{B}$. Interpretations can be extended to formulas:

- **atom** if $\varphi = x_i$, then $\omega(\varphi) = \omega(x_i)$
- **negation** if $\varphi = \neg\varphi_1$, then $\omega(\varphi) = 1 - \omega(\varphi_1)$
- **conjunction** if $\varphi = \varphi_1 \wedge \varphi_2$, then $\omega(\varphi) = \min(\omega(\varphi_1), \omega(\varphi_2))$

Semantics of Propositional Logic

Let $V = \{x_1, \ldots, x_n\}$ be a set of Boolean variables. An interpretation ω is a mapping from V to $\mathbb{B}$. Interpretations can be extended to formulas:

atom if $\varphi = x_i$, then $\omega(\varphi) = \omega(x_i)$

negation if $\varphi = \neg\varphi_1$, then $\omega(\varphi) = 1 - \omega(\varphi_1)$

conjunction if $\varphi = \varphi_1 \land \varphi_2$, then $\omega(\varphi) = \min(\omega(\varphi_1), \omega(\varphi_2))$

disjunction if $\varphi = \varphi_1 \lor \varphi_2$, then $\omega(\varphi) = \max(\omega(\varphi_1), \omega(\varphi_2))$

Semantics of Propositional Logic

Let $V = \{x_1, \ldots, x_n\}$ be a set of Boolean variables. An interpretation ω is a mapping from V to $\mathbb{B}$. Interpretations can be extended to formulas:

- **atom** if $\varphi = x_i$, then $\omega(\varphi) = \omega(x_i)$
- **negation** if $\varphi = \neg\varphi_1$, then $\omega(\varphi) = 1 - \omega(\varphi_1)$
- **conjunction** if $\varphi = \varphi_1 \wedge \varphi_2$, then $\omega(\varphi) = \min(\omega(\varphi_1), \omega(\varphi_2))$
- **disjunction** if $\varphi = \varphi_1 \vee \varphi_2$, then $\omega(\varphi) = \max(\omega(\varphi_1), \omega(\varphi_2))$
- **implication** if $\varphi = \varphi_1 \Rightarrow \varphi_2$, then $\omega(\varphi) = \omega(\neg\varphi_1 \vee \varphi_2)$

Semantics of Propositional Logic

Let $V = \{x_1, \ldots, x_n\}$ be a set of Boolean variables. An interpretation ω is a mapping from V to $\mathbb{B}$. Interpretations can be extended to formulas:

atom if $\varphi = x_i$, then $\omega(\varphi) = \omega(x_i)$

negation if $\varphi = \neg\varphi_1$, then $\omega(\varphi) = 1 - \omega(\varphi_1)$

conjunction if $\varphi = \varphi_1 \wedge \varphi_2$, then $\omega(\varphi) = \min(\omega(\varphi_1), \omega(\varphi_2))$

disjunction if $\varphi = \varphi_1 \vee \varphi_2$, then $\omega(\varphi) = \max(\omega(\varphi_1), \omega(\varphi_2))$

implication if $\varphi = \varphi_1 \Rightarrow \varphi_2$, then $\omega(\varphi) = \omega(\neg\varphi_1 \vee \varphi_2)$

equivalence if $\varphi = \varphi_1 \Leftrightarrow \varphi_2$, then $\omega(\varphi) = \omega((\varphi_1 \Rightarrow \varphi_2) \wedge (\varphi_2 \Rightarrow \varphi_1))$

Semantics of Propositional Logic
Example of Interpretation

$V = \{x_1, x_2, x_3, x_4\}, \omega(x_1) = \omega(x_4) = 0, \omega(x_2) = \omega(x_3) = 1$

- $\omega(\neg x_1) = 1 - \omega(x_1) = 1$, $\omega(\neg x_2) = 1 - \omega(x_2) = 0$

Semantics of Propositional Logic
Example of Interpretation

$V = \{x_1, x_2, x_3, x_4\}, \omega(x_1) = \omega(x_4) = 0, \omega(x_2) = \omega(x_3) = 1$

- $\omega(\neg x_1) = 1 - \omega(x_1) = 1$, $\omega(\neg x_2) = 1 - \omega(x_2) = 0$

Intuitively:

- the negation takes the opposite of the original value

$V = \{x_1, x_2, x_3, x_4\}$, $\omega(x_1) = \omega(x_4) = 0, \omega(x_2) = \omega(x_3) = 1$

- $\omega(\neg x_1) = 1 - \omega(x_1) = 1$, $\omega(\neg x_2) = 1 - \omega(x_2) = 0$
- $\omega(x_1 \land x_2) = \min(\omega(x_1), \omega(x_2)) = \min(0, 1) = 0$

Intuitively:

- the negation takes the opposite of the original value

$V = \{x_1, x_2, x_3, x_4\}, \omega(x_1) = \omega(x_4) = 0, \omega(x_2) = \omega(x_3) = 1$

- $\omega(\neg x_1) = 1 - \omega(x_1) = 1$, $\omega(\neg x_2) = 1 - \omega(x_2) = 0$
- $\omega(x_1 \land x_2) = \min(\omega(x_1), \omega(x_2)) = \min(0, 1) = 0$
- $\omega(x_2 \land x_3) = \min(\omega(x_2), \omega(x_3)) = \min(1, 1) = 1$

Intuitively:

- the negation takes the opposite of the original value

Semantics of Propositional Logic
Example of Interpretation

$V = \{x_1, x_2, x_3, x_4\}, \omega(x_1) = \omega(x_4) = 0, \omega(x_2) = \omega(x_3) = 1$

- $\omega(\neg x_1) = 1 - \omega(x_1) = 1$, $\omega(\neg x_2) = 1 - \omega(x_2) = 0$
- $\omega(x_1 \wedge x_2) = \min(\omega(x_1), \omega(x_2)) = \min(0, 1) = 0$
- $\omega(x_2 \wedge x_3) = \min(\omega(x_2), \omega(x_3)) = \min(1, 1) = 1$

Intuitively:

- the negation takes the opposite of the original value
- the conjunction is true iff both values are true

$V = \{x_1, x_2, x_3, x_4\}, \omega(x_1) = \omega(x_4) = 0, \omega(x_2) = \omega(x_3) = 1$

- $\omega(\neg x_1) = 1 - \omega(x_1) = 1$, $\omega(\neg x_2) = 1 - \omega(x_2) = 0$
- $\omega(x_1 \wedge x_2) = \min(\omega(x_1), \omega(x_2)) = \min(0, 1) = 0$
- $\omega(x_2 \wedge x_3) = \min(\omega(x_2), \omega(x_3)) = \min(1, 1) = 1$
- $\omega(x_1 \vee x_2) = \max(\omega(x_1), \omega(x_2)) = \max(0, 1) = 1$

Intuitively:

- the negation takes the opposite of the original value
- the conjunction is true iff both values are true

Semantics of Propositional Logic

Example of Interpretation

$V = \{x_1, x_2, x_3, x_4\}, \omega(x_1) = \omega(x_4) = 0, \omega(x_2) = \omega(x_3) = 1$

- $\omega(\neg x_1) = 1 - \omega(x_1) = 1$, $\omega(\neg x_2) = 1 - \omega(x_2) = 0$
- $\omega(x_1 \land x_2) = \min(\omega(x_1), \omega(x_2)) = \min(0, 1) = 0$
- $\omega(x_2 \land x_3) = \min(\omega(x_2), \omega(x_3)) = \min(1, 1) = 1$
- $\omega(x_1 \lor x_2) = \max(\omega(x_1), \omega(x_2)) = \max(0, 1) = 1$
- $\omega(x_1 \lor x_4) = \max(\omega(x_1), \omega(x_4)) = \max(0, 0) = 0$

Intuitively:

- the negation takes the opposite of the original value
- the conjunction is true iff both values are true

Semantics of Propositional Logic

Example of Interpretation

$V = \{x_1, x_2, x_3, x_4\}, \omega(x_1) = \omega(x_4) = 0, \omega(x_2) = \omega(x_3) = 1$

- $\omega(\neg x_1) = 1 - \omega(x_1) = 1, \omega(\neg x_2) = 1 - \omega(x_2) = 0$
- $\omega(x_1 \wedge x_2) = \min(\omega(x_1), \omega(x_2)) = \min(0, 1) = 0$
- $\omega(x_2 \wedge x_3) = \min(\omega(x_2), \omega(x_3)) = \min(1, 1) = 1$
- $\omega(x_1 \vee x_2) = \max(\omega(x_1), \omega(x_2)) = \max(0, 1) = 1$
- $\omega(x_1 \vee x_4) = \max(\omega(x_1), \omega(x_4)) = \max(0, 0) = 0$

Intuitively:

- the negation takes the opposite of the original value
- the conjunction is true iff both values are true
- the disjunction is true iff at least one of the values is true

Other Notations for Interpretations

Sets of Atoms

- An interpretation can be written as the set of atoms which are assigned 1

Other Notations for Interpretations

Sets of Atoms

- An interpretation can be written as the set of atoms which are assigned 1
- Example: $V = \{x_1, x_2, x_3\}, \omega(x_1) = 0, \omega(x_2) = \omega(x_3) = 1$ is equivalent to $\omega = \{x_2, x_3\}$

Other Notations for Interpretations

Sets of Atoms

- An interpretation can be written as the set of atoms which are assigned 1
- Example: $V = \{x_1, x_2, x_3\}, \omega(x_1) = 0, \omega(x_2) = \omega(x_3) = 1$ is equivalent to $\omega = \{x_2, x_3\}$

Vectors of Bits

- An interpretation can be written as a vector of 0 and 1; the order has to be fixed (usually lexicographical order)

Other Notations for Interpretations

Sets of Atoms

- An interpretation can be written as the set of atoms which are assigned 1
- Example: $V = \{x_1, x_2, x_3\}, \omega(x_1) = 0, \omega(x_2) = \omega(x_3) = 1$ is equivalent to $\omega = \{x_2, x_3\}$

Vectors of Bits

- An interpretation can be written as a vector of 0 and 1; the order has to be fixed (usually lexicographical order)
- Example: $V = \{x_1, x_2, x_3\}, \omega(x_1) = 0, \omega(x_2) = \omega(x_3) = 1$ is equivalent to $\omega = 011$

Satisfaction of Formulas

- If $\omega(\varphi) = 1$, then ω is a **model** of φ. We also say that φ is **satisfied** by ω
- If $\omega(\varphi) = 0$, then ω is a counter-model of φ
- $\mathsf{mod}(\varphi)$ is the set of models of φ
- If $\mathsf{mod}(\varphi) = \emptyset$, then φ is **inconsistent** (or **insatisfiable**)
- If $\mathsf{mod}(\varphi) \neq \emptyset$, then φ is **consistent** (or **satisfiable**)
- If $\mathsf{mod}(\varphi)$ is the set of all possible interpretations, then φ is **valid**

Satisfaction, Consequence, Equivalence

We define some meta-language symbols for reasoning about interpretations and formulas:

- $\omega \models \varphi$ means that (the interpretation) ω satisfies (the formula) φ, *i.e.* $\omega \in \mathtt{mod}(\varphi)$
- $\varphi \vdash \psi$ means that (the formula) ψ is a consequence of (the formula) φ, formally defined as $\mathtt{mod}(\varphi) \subseteq \mathtt{mod}(\psi)$
- $\varphi \equiv \psi$ means that φ and ψ are equivalent, formally defined as $\varphi \vdash \psi$ and $\psi \vdash \varphi$ (which implies $\mathtt{mod}(\varphi) = \mathtt{mod}(\psi)$)

Normal Forms

- A formula is a **literal** iff it is an atom or the negation of an atom
  x_1 and $\neg x_1$ are literals; $x_1 \vee x_2$ is not a literal

Normal Forms

- A formula is a **literal** iff it is an atom or the negation of an atom
  x_1 and $\neg x_1$ are literals; $x_1 \vee x_2$ is not a literal
- A formula is a **clause** iff it is a disjunction of literals
  $x_1 \vee x_2 \vee \neg x_3$ is a clause ; $x_1 \vee (x_2 \wedge x_3)$ is not a clause

Normal Forms

- A formula is a **literal** iff it is an atom or the negation of an atom
  x_1 and $\neg x_1$ are literals; $x_1 \vee x_2$ is not a literal
- A formula is a **clause** iff it is a disjunction of literals
  $x_1 \vee x_2 \vee \neg x_3$ is a clause ; $x_1 \vee (x_2 \wedge x_3)$ is not a clause
- A formula is a **cube** iff it is a conjunction of literals
  $x_1 \wedge x_2 \wedge \neg x_3$ is a cube ; $x_1 \wedge (x_2 \vee x_3)$ is not a cube

Normal Forms

- A formula is a **literal** iff it is an atom or the negation of an atom
  x_1 and $\neg x_1$ are literals; $x_1 \vee x_2$ is not a literal
- A formula is a **clause** iff it is a disjunction of literals
  $x_1 \vee x_2 \vee \neg x_3$ is a clause ; $x_1 \vee (x_2 \wedge x_3)$ is not a clause
- A formula is a **cube** iff it is a conjunction of literals
  $x_1 \wedge x_2 \wedge \neg x_3$ is a cube ; $x_1 \wedge (x_2 \vee x_3)$ is not a cube

Conjunctive Normal Form

A formula is in Conjunctive Normal Form (CNF) iff it is a conjunction of clauses

Normal Forms

- A formula is a **literal** iff it is an atom or the negation of an atom
  x_1 and $\neg x_1$ are literals; $x_1 \vee x_2$ is not a literal
- A formula is a **clause** iff it is a disjunction of literals
  $x_1 \vee x_2 \vee \neg x_3$ is a clause ; $x_1 \vee (x_2 \wedge x_3)$ is not a clause
- A formula is a **cube** iff it is a conjunction of literals
  $x_1 \wedge x_2 \wedge \neg x_3$ is a cube ; $x_1 \wedge (x_2 \vee x_3)$ is not a cube

Conjunctive Normal Form

A formula is in Conjunctive Normal Form (CNF) iff it is a conjunction of clauses

Disjunctive Normal Form

A formula is in Disjunctive Normal Form (DNF) iff it is a disjunction of cubes

Examples of CNF formulas

- $(x_1 \vee x_2) \wedge (x_3 \vee x_4)$
- $(x_1 \vee \neg x_2 \vee x_5) \wedge (x_3 \vee x_5)$

Examples of DNF formulas

- $(x_1 \wedge x_2) \vee (x_3 \wedge x_4)$
- $x_1 \vee (x_2 \wedge x_4) \vee (x_3 \wedge x_4 \wedge \neg x_5)$

- A clause is satisfied if at least one of its literals is satisfied
- A CNF formula is satisfied if all its clauses are satisfied

Example

With $\omega(x_1) = \omega(x_2) = 1$ and $\omega(x_3) = \omega(x_4) = 0$

- $(x_1 \vee x_2) \wedge (x_3 \vee x_4)$ is not satisfied

- A clause is satisfied if at least one of its literals is satisfied
- A CNF formula is satisfied if all its clauses are satisfied

Example

With $\omega(x_1) = \omega(x_2) = 1$ and $\omega(x_3) = \omega(x_4) = 0$

- $(x_1 \vee x_2) \wedge (x_3 \vee x_4)$ is not satisfied

With $\omega(x_1) = \omega(x_3) = 1$ and $\omega(x_2) = \omega(x_4) = 0$

- $(x_1 \vee x_2) \wedge (x_3 \vee x_4)$ is satisfied

- A cube is satisfied if all its literals are satisfied
- A DNF formula is satisfied if at least one of its cubes is satisfied

Example

With $\omega(x_1) = \omega(x_3) = 1$ and $\omega(x_2) = \omega(x_4) = 0$

- $(x_1 \land x_2) \lor (x_3 \land x_4)$ is not satisfied

- A cube is satisfied if all its literals are satisfied
- A DNF formula is satisfied if at least one of its cubes is satisfied

Example

With $\omega(x_1) = \omega(x_3) = 1$ and $\omega(x_2) = \omega(x_4) = 0$

- $(x_1 \land x_2) \lor (x_3 \land x_4)$ is not satisfied

With $\omega(x_1) = \omega(x_2) = 1$ and $\omega(x_3) = \omega(x_4) = 0$

- $(x_1 \land x_2) \lor (x_3 \land x_4)$ is satisfied