

離散構造

筑波大学 情報学群 情報科学類

はじめに

離散構造 (Discrete Structures) は、その名の通り、離散的な構造のことであり、通常の数学 (解析, 幾何, 代数など) が主として連続系を対象としているのに対して、必ずしも連続的でない構造を持つものの総称である¹。

現代のコンピュータ²はハードウェアのみならずソフトウェアも 0/1 の二値に代表される離散的な構造を基礎として構成されているため、現代コンピュータ科学を理解するために必要な数学の多くは、離散的な数学 (離散数学) である。

このテキストは、離散構造の代表的なものである論理, 集合, 関係, 関数, グラフに対する入門書である。

このテキストの読み方と注意事項

- このテキストは、基本的な内容と、発展的な内容とから構成されている。各章の題目のあとに (†) の記号がついているものは、発展的な内容であり、余力がある者が進んだ内容を学習するためのものである。(†) の記号がついていない章は、全ての者が学習すべき内容を表す。
- 数学的な議論では、事実を主張するためには必ず厳密な証明を必要とする。証明においては、何を仮定して、何を導き出そうとしているかを常に確認すること。しばしば「証明したいことを仮定している」ような誤りが見受けられる。また、どのような定義、定理、推論規則を使って証明しているのかを意識する必要がある。「自分はそのことを事実として知っている」や「当たり前」では証明にならない。
- このテキストに、誤りや不適切な説明等がある場合は、担当教官 または 担当 Teaching Assistant に連絡してほしい。
- このテキストは、『離散構造』の講義のために作成したものであり、許可なく、目的外利用・複製・再配布等をしてはいけません。

¹離散/連続の分類は対象ごとに厳格に決まっているものではなく、対象を見る立場に依存することがある。この授業でも、集合の例として実数全体の集合を考えたり、連続関数全体の集合を考えることがある。

²将来のコンピュータがすべて離散的な原理で構成されているとは限らない。たとえば、現在活発に研究されている量子コンピュータは連続的な確率分布が基礎にある。

目次

第 1 章 論理	1
1.1 非論理的な思考	1
1.2 命題と真理値	1
1.3 命題を構成する方法	2
1.3.1 「でない」	2
1.3.2 「かつ」と「または」	2
1.3.3 「ならば」	3
1.3.4 「同値」	3
1.4 命題の例	3
1.5 括弧 (かっこ) の用法	4
1.6 逆と対偶	4
1.7 必要条件と十分条件	5
1.8 基本的な証明技法	5
1.8.1 すべての場合をチェックする (しらみつぶし法)	5
1.8.2 変数を使う	5
1.8.3 「ならば」の鎖	5
1.8.4 間接法	6
1.8.5 $A \Leftrightarrow B$ の証明	7
1.8.6 避けるべき証明法	7
1.9 「すべて」と「ある」	7
1.9.1 「すべて」	7
1.9.2 「ある」	8
1.9.3 暗黙の全称記号	8
1.9.4 進んだ例 (†)	8
1.9.5 全称記号・存在記号を含む命題 (†)	9
1.9.6 全称, 存在に関する命題の証明	10
1.9.7 さらに進んだ学習のために	10
第 2 章 集合	11
2.1 集合の構成と表現	11
2.1.1 要素を一つ一つ書き並べる方法	11
2.1.2 要素が満たすべき性質から集合を作る方法—内包的な表現	11
2.1.3 内包的な表現の発展	12
2.2 集合の等しさ	12
2.3 集合の包含関係	13
2.4 集合に関する証明法	13
2.4.1 $A \subset B$ の証明法	13
2.4.2 $A \not\subset B$ の証明法	13

2.4.3	$A = B$ の証明法	13
2.4.4	鳩の巣原理 (Pigeon hole principle) (†)	14
2.5	集合の演算	14
2.5.1	べき集合	14
2.5.2	和集合と共通部分	14
2.5.3	差集合	16
2.5.4	補集合 (†)	16
2.5.5	組と直積集合	17
2.6	可算集合と対角線論法 (†)	18
第 3 章	関数	21
3.1	定義域と値域	21
3.2	複数の引数をもつ関数	22
3.3	像 (image)	22
3.4	逆像 (inverse image) (†)	22
3.5	関数の等しさ	23
3.6	関数の合成 (composition)	23
3.7	恒等関数 (identity function)	24
3.8	単射 (一对一写像, one to one, injection)	24
3.9	全射 (上への写像, onto, surjection)	25
3.10	全単射 (bijection) と逆関数 (inverse function)	25
3.11	部分関数 (partial function)	25
第 4 章	関係	27
4.1	関係	27
4.2	二項関係の性質	27
4.3	順序	28
4.4	同値関係	29
4.5	関係の合成	30
4.6	閉包 (closure) (†)	31
第 5 章	グラフと木	34
5.1	グラフ	34
5.1.1	無向グラフ	34
5.1.2	有向グラフ	34
5.1.3	位数とサイズ	35
5.1.4	道 (パス) と閉路 (サイクル)	35
5.1.5	グラフの同型	36
5.1.6	部分グラフ	36
5.1.7	連結, 連結成分 (†)	37
5.1.8	完全グラフ, 完全 2 部グラフ (†)	37
5.1.9	グラフの応用 (†)	37
5.2	木 (tree)	38
5.2.1	順序木	39
5.2.2	走査	40

第 6 章	帰納的定義と帰納法	42
6.1	帰納的に定義された集合	42
6.1.1	リスト	43
6.1.2	文字列	44
6.1.3	2 分木 (binary Tree)	46
6.1.4	BNF 記法 ($\dagger$)	48
6.2	帰納的に定義された関数	48
6.3	帰納法による証明	51
6.3.1	数学的帰納法	51
6.3.2	リストに関する帰納法	53
6.3.3	2 分木に関する帰納法 ($\dagger$)	54

た