Algorithmic graph theory
ALAN GIBBONS
Department of Computer Science, University of Warwick

Algorithmic graph theory
Published by the Press Syndicate of the University of Cambridge
The Pitt Building, Trumpington Street, Cambridge CB2 1RP
40 West 20th Street, New York, NY 10011-4211, USA
10 Stamford Road, Oakleigh, Melbourne 3166, Australia

© Cambridge University Press 1985
First published 1985

Library of Congress catalogue card number: 84-23835
British Library cataloguing in publication data
Gibbons, Alan
Algorithmic graph theory.
1. Graph theory 2. Algorithms
1. Title
511'.5 QA166

ISBN 0 521 28881 9 paperback

Transferred to digital printing 1999
To my children
Gabrielle, Chantal and Rosalind,
with love.
Contents

Preface (xi)

1 Introducing graphs and algorithmic complexity 1
   1.1 Introducing graphs 1
   1.2 Introducing algorithmic complexity 8
   1.3 Introducing data structures and depth-first searching 16
      1.3.1. Adjacency matrices and adjacency lists 17
      1.3.2. Depth-first searching 20
      1.3.3. Two linear-time algorithms 24
   1.4 Summary and references 32
      Exercises 33

2 Spanning-trees, branchings and connectivity 39
   2.1 Spanning-trees and branchings 39
      2.1.1. Optimum weight spanning-trees 40
      2.1.2. Optimum branchings 42
      2.1.3. Enumeration of spanning-trees 49
   2.2 Circuits, cut-sets and connectivity 54
      2.2.1. Fundamental circuits of a graph 54
      2.2.2. Fundamental cut-sets of a graph 57
      2.2.3. Connectivity 60
   2.3 Summary and references 62
      Exercises 63

3 Planar graphs 67
   3.1 Basic properties of planar graphs 67
   3.2 Genus, crossing-number and thickness 71
   3.3 Characterisations of planarity 75
      3.3.1. Dual graphs 81
   3.4 A planarity testing algorithm 85
   3.5 Summary and references 92
      Exercises 93
viii Contents

4 Networks and flows 96
4.1 Networks and flows 96
4.2 Maximising the flow in a network 98
4.3 Menger's theorems and connectivity 106
4.4 A minimum-cost flow algorithm 111
4.5 Summary and references 118
   Exercises 120

5 Matchings 125
5.1 Definitions 125
5.2 Maximum-cardinality matchings 126
   5.2.1. Perfect matchings 134
5.3 Maximum-weight matchings 136
5.4 Summary and references 147
   Exercises 148

6 Eulerian and Hamiltonian tours 153
6.1 Eulerian paths and circuits 153
   6.1.1. Eulerian graphs 155
   6.1.2. Finding Eulerian circuits 156
6.2 Postman problems 161
   6.2.1. Counting Eulerian circuits 162
   6.2.2. The Chinese postman problem for undirected graphs 163
   6.2.3. The Chinese postman problem for digraphs 165
6.3 Hamiltonian tours 169
   6.3.1. Some elementary existence theorems 169
   6.3.2. Finding all Hamiltonian tours by matricial products 173
   6.3.3. The travelling salesman problem 175
   6.3.4. 2-factors of a graph 182
6.4 Summary and references 184
   Exercises 185

7 Colouring graphs 189
7.1 Dominating sets, independence and cliques 189
7.2 Colouring graphs 195
   7.2.1. Edge-colouring 195
   7.2.2. Vertex-colouring 198
   7.2.3. Chromatic polynomials 201
7.3 Face-colourings of embedded graphs 204
   7.3.1. The five-colour theorem 204
   7.3.2. The four-colour theorem 207
7.4 Summary and references 210
   Exercises 212

8 Graph problems and intractability 217
8.1 Introduction to NP-completeness 217
Contents

8.1.1. The classes $P$ and $NP$ .......................... 217
8.1.2. $NP$-completeness and Cook’s theorem .......... 222
8.2 $NP$-complete graph problems ....................... 227
  8.2.1. Problems of vertex cover, independent set and clique 227
  8.2.2. Problems of Hamiltonian paths and circuits and the travelling salesman problem 229
  8.2.3. Problems concerning the colouring of graphs .... 235
8.3 Concluding comments .................................. 241
8.4 Summary and references ............................. 244
   Exercises ............................................. 245
Appendix: On linear programming ...................... 249
Author index ............................................. 254
Subject index ............................................. 256
Preface

In the last decade or so work in graph theory has centred on algorithmic interests rather than upon existence or characterisation theorems. This book reflects that change of emphasis and is intended to be an introductory text for undergraduates or for new postgraduate students.

The book is aimed primarily at computer scientists. For them graph theory provides a useful analytical tool and algorithmic interests are bound to be uppermost. The text does, however, contain an element of traditional material and it is quite likely that the needs of a wider audience, including perhaps mathematicians and engineers, will be met. Hopefully, enough of this material has been included to suggest the mathematical richness of the field.

Prerequisites for an understanding of the text have been kept to a minimum. It is essential however to have had some exposure to a high-level, procedural and preferably recursive programming language, to be familiar with elementary set notation and to be at ease with (for example, inductive) theorem proving. Where more advanced concepts are required the text is largely self-contained. This is true, for example, in the use of linear programming and in the proofs of $NP$-completeness.

There is rather more material than would be required for a one-semester course. It is possible to use the text for courses of more or of less difficulty, or to select material as it appeals. For example an elementary course might not include, amongst other material, that on branchings (in chapter 2), minimum-cost flows (in chapter 4), maximum-weight matchings (in chapter 5), postman problems (in chapter 6) and proofs of $NP$-completeness (all of chapter 8). Whatever the choice of material, any course will inevitably reflect the main preoccupation of the text. This is to identify those important problems in graph theory which have an efficient algorithmic solution (that is, those whose time-complexity is polynomial in the problem
Preface

size) and those which, it is thought, do not. In this endeavour the most
efficient of the known polynomial time algorithms have not necessarily
been described. These algorithms can require explanations that are too
lengthy and may have difficult proofs of correctness. One such example is
graph planarity testing in linear-time. It has been thought preferable to go
for breadth of material and, where required, to provide references to more
difficult and stronger results. Nevertheless, a body of material and quite a
few results, which are not easily available elsewhere, have been presented
in elementary fashion.

The exercises which appear at the ends of chapters often extend or
motivate the material of the text. For this reason outlines of solutions are
invariably included. Some benefit can certainly be obtained by reading
these sections even if detailed solutions are not sought.

Thanks are due to Valerie Gladman for her cheerful typing of the manu-
script. Primary and secondary sources of material are referenced at the ends
of chapters. I gratefully acknowledge my debt to the authors of these works.
However, I claim sole responsibility for any obscurities and errors that
the text may contain.

A. M. Gibbons Warwick, January 1984