

CS_232 Algorithms and Complexity

Week 1, Monday, 14/1/2008

Oliver Kullmann

Department of Computer Science
Swansea University
Swansea, SA2 8PP, UK

e-mail: O.Kullmann@Swansea.ac.uk

<http://cs.swan.ac.uk/~csoliver>

December 16, 2008

Lecture held on Monday, January 14, 2008, in the Glyndwr Building, Lecture Hall C, from 13:00 - 14:00.

I Solutions for Coursework II

I Graph colouring

Bipartite graphs

1) *Determine the chromatic number of graphs from the following graph classes, and prove your answer:*

- the complete graphs K_n for $n \geq 2$, where one edge has been removed;*
- forests.*

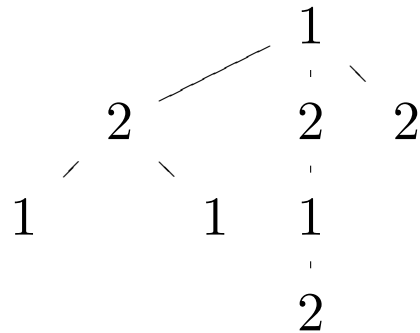
Consider $n \geq 0$. The chromatic number of K_n is $\chi(K_n) = n$, since this graph is complete. Now assume $n \geq 2$, consider an edge $e \in E(K_n)$, and let $G' := K_n - e = (V(K_n), E(K_n) \setminus \{e\})$. That is, G' is obtained from K_n by removing the edge e , while keeping all vertices. The question is to determine $\chi(G')$. To help us providing the details, let e have endpoints a, b (that is, $e = \{a, b\}$).

We have $\chi(G') = n - 1$, which is proven as follows:

- First we need to show $\chi(G) \leq n-1$, that is, we need to provide a colouring of G using only $n-1$ colours. This is achieved by giving the two endpoints a, b of e the same colour, while giving every other vertex a different colour: So we use $1 + (n-2) = n-1$ colours. More precisely, we can specify the colouring by recalling that $V(K_n) = V(G) = \{1, \dots, n\}$, so let the colour of the a, b be 0, while every vertex in $V(G) \setminus \{a, b\}$ is coloured by itself.
- Additionally we need to show $\chi(G) \geq n-1$, that is, we need to show that less colours are not sufficient. Assume that f is a colouring of G using at most $n-2$ colours, Consider the set of vertices $V' := V(G) \setminus \{a\}$: This set has $|V'| = n-1$ members, more than colours, so there are $x, y \in V'$, $x \neq y$, which get the same colour (i.e., $f(x) = f(y)$). Now x, y are adjacent in G , since neither x nor y is a , but the edge $\{a, b\}$ is the only missing edge.

Now to the forests. First we observe that we only need to consider the connected components of forests, the trees, since the connected components can be

coloured independently. So we need to compute the chromatic numbers of trees. For example



To make things simpler, we have *drawn* here the tree as a rooted tree. And actually, this drawing also helps understanding the colouring scheme: Considering the paths from the root to some vertices, colour the vertices on the paths alternatingly 1 and 2. To make this fully precise would need a bit more work, but fortunately we have proven the following general theorem:

A graph G is not bipartite if and only if G has a circuit of odd length.

Now trees (and more generally, forests) do not have any circuits, and thus they are bipartite. More precisely we have $\chi(G) \leq 2$ for a forest G , where $\chi(G) = 0$ iff G has no vertex, and $\chi(G) = 1$ iff G has no edge.

Greedy Colouring

2) For a graph G denote by $\Delta(G)$ the maximal vertex degree.

1. Show that $\chi(G) \leq \Delta(G) + 1$.
2. Can we also (efficiently) find such a colouring?
3. And are there graphs with $\chi(G) = \Delta(G) + 1$?

Solution:

Running GCA on a graph G for *any* vertex order yields a colouring using at most $\Delta(G) + 1$ colours:

The worst case for the algorithm is given when encountering a vertex v having the maximal degree $\Delta(G)$, where all vertices are already coloured, and this by different colours — here colour $\Delta(G) + 1$ has to be used for v , but that suffices.

Thus we can find such a colouring in linear time. Examples where the bound $\chi(G) \leq \Delta(G) + 1$ is sharp:

1. The complete graphs K_n for $n \in \mathbb{N}_0$ (we have $\chi(G) = n$ and $\Delta(G) = n - 1$).
2. The odd cycles C^k for odd $k \geq 3$.

Interestingly, these two examples are the only examples due to the following theorem of Brooks:

Theorem *If a graph G is complete or (isomorphic to) an odd cycle, then we have $\chi(G) = \Delta(G) + 1$, while otherwise we have $\chi(G) \leq \Delta(G)$ (and we can also find such colourings efficiently).*

Final remark on greedy colouring: Always assuming the worst case for greedy colouring, that all coloured neighbours of the vertex to be coloured have different colours, the smallest number of colours greedy colouring could come up with for G would be the minimal $k \in \mathbb{N}_0$ such that there exists an order v_1, \dots, v_n of the vertices of G with the property, that for every $i \in \{1, \dots, n\}$ the number of neighbours of v_i among $\{v_1, \dots, v_{i-1}\}$ is strictly less than k . This k is called the *colouring number* $\text{col}(G) \in \mathbb{N}_0$. We have

$$\chi(G) \leq \text{col}(G) \leq \Delta(G) + 1$$

for any graph G . An example for $\chi(G) < \text{col}(G)$ is given by an even cycle C^k ($k \geq 4$, k even), where $\text{col}(C^k) = 3$, while $\chi(C^k) = 2$.

Algorithms for graph colouring

3) *Design an “intelligent” algorithm (in pseudo-code) for solving the k -colouring problem (whether a graph is k -colourable or not), and discuss the potential savings compared to the algorithm simply enumerating all possible (potential) colourings.*

The natural solution is given by a backtracking algorithm, which recursively tries to extend a partial colouring (colouring only some nodes, initially none) to a complete colouring. The savings come from realising as early as possible that a given partial colouring can not be extended.

So the minimum of intelligence our algorithm should have is to check, when assigning a new colour, which edges are now completely coloured, and to check that all those edges fulfil the colouring condition (their endpoints are coloured differently).

Another important speed-up, actually including that minimal check, comes from maintaining for each vertex w the set $C(w)$ of remaining possible

colours (not yet used by any neighbour; initially $C(w) = \{1, \dots, k\}$): When colouring a vertex v with colour i , we need to visit the neighbours w of v , and remove i from $C(w)$. Now if $C(w) = \emptyset$, then we can immediately backtrack. And furthermore if $C(w)$ only contains one element, i.e., $C(w) = \{j\}$, then we can colour w immediately without backtracking!

(Notice that this strengthened technique solves every cycle graph with only one backtrack (for the initial vertex), when $k = 2$. And with a bit more care we can make sure that always (for every k) every bipartite graph is coloured using (at most) two colours.)

II The Euclidean algorithm

1) *Compute the Euclidean sequence and the Euclidean extension sequence for the following pairs of numbers a, b ; in each case state explicitly how to represent $\gcd(a, b)$ as a linear combination of a and b :*

1. 66, 55
2. 113, 63
3. 255, 187
4. 104, 116
5. 1001, 338.

Solution:

1. 66, 55:

(a) 66, 55, 11, 0

(b) (1, 0), (0, 1), (1, -1), (-5, 6)

(c) $\gcd(66, 55) = 11 = 1 \cdot 66 + (-1) \cdot 55$.

2. 113, 63:

(a) 113, 63, 50, 13, 11, 2, 1, 0

(b) $(1, 0), (0, 1), (1, -1), (-1, 2), (4, -7), (-5, 9),$
 $(29, -52), (-63, 113)$

(c) $\gcd(113, 63) = 1 = 29 \cdot 113 + (-52) \cdot 63.$

3. 255, 187:

(a) 255, 187, 68, 51, 17, 0

(b) $(1, 0), (0, 1), (1, -1), (-2, 3), (3, -4), (-11, 15)$

(c) $\gcd(255, 187) = 17 = 3 \cdot 255 + (-4) \cdot 187.$

4. 104, 116:

(a) 104, 116, 104, 12, 8, 4, 0

(b) $(1, 0), (0, 1), (1, 0), (-1, 1), (9, -8), (-10, 9),$
 $(29, -26)$

(c) $\gcd(104, 116) = 4 = (-10) \cdot 104 + 9 \cdot 116.$

5. 1001, 338:

(a) 1001, 338, 325, 13, 0

(b) $(1, 0), (0, 1), (1, -2), (-1, 3), (26, -77)$

(c) $\gcd(1001, 338) = 13 = (-1) \cdot 1001 + 3 \cdot 338.$

III Modular arithmetic

Basics

1) *Explain how modular addition, subtraction and multiplication works, showing also some example calculations.*

XXX

Some examples:

1. $23 +_{13} 45 = 3 \in \mathbb{Z}_{13}$

2. $133 *_{55} 68 = 24 \in \mathbb{Z}_{55}$

3. $6 -_{17} 22 = 1 \in \mathbb{Z}_{17}$.

Multiplication tables

2) Assume that the multiplication table of \mathbb{Z}_n is given. How then can we easily determine elements which are invertible (multiplicatively), and find their inverses in the positive cases?

Consider the case $n = 7$. The addition table is (shown for completeness):

$+_7$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	2	3	4	5	6	0
2	2	3	4	5	6	0	1
3	3	4	5	6	0	1	2
4	4	5	6	0	1	2	3
5	5	6	0	1	2	3	4
6	6	0	1	2	3	4	5

And the multiplication table is:

$*_7$	0	1	2	3	4	5	6
0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6
2	0	2	4	6	1	3	5
3	0	3	6	2	5	1	4
4	0	4	1	5	2	6	3
5	0	5	3	1	6	4	2
6	0	6	5	4	3	2	1

XXX

- Note the symmetry of both tables (addition and multiplication are commutative).
- Note that writing down the addition table is trivial, and creating the multiplication table is greatly simplified by the observation, that the next entry in a row is obtained by modularly adding the leading row entry (i.e., the row label)!
- The addition table is a “Latin square” (every row and every column is a permutation of $\{0, 1, \dots, 6\}$); this is true for the addition table of every \mathbb{Z}_n .
- 7 is prime, thus also the multiplication table is a Latin square after removal of first row and column!

Now we consider invertibility in \mathbb{Z}_m . Speaking of “invertibility” in \mathbb{Z}_m always means “multiplicative invertibility”, since every element is additively invertible. Having the multiplication table of \mathbb{Z}_m at hand, it is easy to determine invertible elements and their inverses: For a given element x , determine if the row of x has an entry equal 1 — if not, then x is not invertible, while otherwise the column element corresponding to the (unique) entry 1 is the inverse.

The smallest n with non-invertible elements (other than 0, of course) is $n = 4$, which has the multiplication table

$*_4$	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	0	2
3	0	3	2	1

We see that the invertible

elements are 1 and 3, that is, $\mathbb{Z}_4^* = \{1, 3\}$.

XXX

Inversion

3) *Decide, which of the following elements are invertible, and if they are, then show the inverse and how to compute it:*

1. 7 in \mathbb{Z}_{17}

2. 18 in \mathbb{Z}_{38}

3. 121 in \mathbb{Z}_{158}

4. 10 in \mathbb{Z}_{19}

5. 4008 in \mathbb{Z}_{4009} .

Solution:

The Euclidean algorithm yields a fast algorithm for deciding invertibility, while with the help of the extended Euclidean algorithm we can also compute inverses.

1. $\gcd(17, 7) = 1 = (-2) \cdot 17 + 5 \cdot 7$, and thus 5 is the inverse of 7 in \mathbb{Z}_{17} . Check $5 * 7 = 35 = 2 \cdot 17 + 1$.

2. $\gcd(38, 18) = 2$, and thus 18 is not invertible in \mathbb{Z}_{38} .
3. $\gcd(158, 121) = 1 = 36 \cdot 158 + (-47) \cdot 121$, and thus $-47 + 158 = 111$ is the inverse of 121 in \mathbb{Z}_{158} .
Check: $111 \cdot 121 = 13431 = 85 \cdot 158 + 1$.
4. $\gcd(19, 10) = 1 = (-1) \cdot 19 + 2 \cdot 10$, and thus 2 is the inverse of 10 in \mathbb{Z}_{19} . Check: $2 \cdot 10 = 20 = 1 \cdot 19 + 1$.
5. 4008 equals -1 “modulo 4009”, and the multiplicative inverse of -1 is -1 (itself); in other words, 4008 is self-inverse in \mathbb{Z}_{4009} . Check: $4008 \cdot 4008 = 16064064 = 4007 \cdot 4009 + 1$.

Modular exponentiation

4) *Compute the following exponentiations (show your calculations; use a pocket calculator; hint: not in all cases the binary expansion of the exponent is actually needed):*

1. $\text{pow}_3(2, 36291928392133)$

2. $\text{pow}_{16}(7, 105)$

3. $\text{pow}_{100000}(20, 10000000)$

4. $\text{pow}_{119}(64, 238)$

5. $\text{pow}_{131}(97, 11401)$.

1. $\text{pow}_3(2, 36291928392133) = 2$; this can be seen directly because 2 equals -1 modulo 3, and -1 to the power of an odd number is -1 .

You also see it easily when performing the algorithm for fast exponentiation: after the first squaring you obtain 1, and that doesn't change anymore, so only the first bit of the binary expansion of the

exponent is needed (0 if the exponent is even, 1 if the exponent is odd; and this bit governs whether whether $2^{[0]} = 2$ is used in the final product or not).

2. $\text{pow}_{16}(7, 105) = 7$

3. $\text{pow}_{100000}(20, 10000000) = 0$

This can also be seen directly: $20 = 2 \cdot 10$ and $10^5 = 100000$, thus already $\text{pow}_{100000}(20, 5) = 0$.

4. $\text{pow}_{119}(64, 238) = 50$.

5. $\text{pow}_{131}(97, 11401) = 42$.

IV Cryptography

In the following we assume $p = 59$, $q = 67$, and thus $n = 59 \cdot 67 = 3953$ and $N = \varphi(n) = 58 \cdot 66 = 3828$. As encryption key we use $e = 1117$.

- 1. Encrypt the plaintext message $m = 3600$, and show that decryption gives back the original message (show your computations).*
- 2. Decrypt the ciphertext message $c = 2566$, and show that encryption of the resulting plaintext gives back the ciphertext (show your computations).*
- 3. Discuss possibilities how to break RSA.*

Solution: For the encryption we have to compute

$$\begin{aligned} \text{RSA}_{n,e}(m) &= \text{RSA}_{3953,1117}(3600) = \\ &= \text{pow}_{3953}(3600, 1117) = 3482. \end{aligned}$$

To decrypt the ciphertext $c = 3482 \in \mathbb{Z}_{3953}$, we have to compute the secret exponent d , the inverse of e in $\mathbb{Z}_N = \mathbb{Z}_{3828}$. As before we compute $\gcd(3828, 1117) = 1 = (-466) \cdot 3828 + 1597 \cdot 1117$, and thus $d = 1597$ is the inverse of e in \mathbb{Z}_N (check: $1597 \cdot 1117 = 1783849 = 466 \cdot 3828 + 1$).

Now we have

$$\begin{aligned} \text{RSA}_{n,e}^{-1}(c) &= \text{RSA}_{3953,1117}^{-1}(3482) = \\ &= \text{pow}_{3953}(3482, 1597) = 3600. \end{aligned}$$

Since we have already computed the secret key d , the second part of the exercise follows suite:

$$\begin{aligned} \text{RSA}_{n,e}^{-1}(c) &= \text{RSA}_{3953,1117}^{-1}(2566) = \\ &= \text{pow}_{3953}(2566, 1597) = 74, \end{aligned}$$

and

$$\begin{aligned} \text{RSA}_{n,e}(m) &= \text{RSA}_{3953,1117}(74) = \\ &= \text{pow}_{3953}(74, 1117) = 2566. \end{aligned}$$

Regarding the (theoretical) possibilities for breaking RSA see Lecture 08a. Though not proven, the only possibilities seems to be to factorise n , that is, to find the (secret) prime numbers p and q with $p \cdot q = n$.

For doing so many sophisticated algorithms have been developed. For small n the “Sieve of Eratosthenes” is most efficient: Run through all the prime numbers $2 \leq p < \sqrt{n}$ and check whether $p \mid n$.

For the above n we have $\sqrt{n} = \sqrt{3953} = 62.87\dots$, and thus we have to check the prime numbers

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31,

37, 41, 43, 47, 53, 59, 61.