MATH 322 – Graph Theory
Fall Term 2021

Notes for Lecture 23

Thursday, December 2

A real-life problem that can be answered via Graph Theory

Scheduling exams: Assume that the Registrar's Office of our university wants to come up with the final exam schedule for the Fall term.

The main restriction that they have to pay attention to is that classes which have common students enrolled should not have their exams scheduled at the same time.

Otherwise, if the student rosters of two courses have an empty intersection, then the final exams of these two courses can take place concurrently (and of course it would be desirable to have multiple exams running at the same time in order to have only a few days of final exams).

A real-life problem that can be answered via Graph Theory

Scheduling exams: Assume that the Registrar's Office of our university wants to come up with the final exam schedule for the Fall term.

The main restriction that they have to pay attention to is that classes which have common students enrolled should not have their exams scheduled at the same time.

Otherwise, if the student rosters of two courses have an empty intersection, then the final exams of these two courses can take place concurrently (and of course it would be desirable to have multiple exams running at the same time in order to have only a few days of final exams).

If we consider a graph with vertex set all the different courses offered in the Fall term, and we assume that there is an edge joining two courses **if and only if their student rosters have a non-empty intersection**, what kind of vertex subsets would we be interested in, that would correspond to groups of courses whose exams can run at the same time?

A real-life problem that can be answered via Graph Theory

Scheduling exams: Assume that the Registrar's Office of our university wants to come up with the final exam schedule for the Fall term.

The main restriction that they have to pay attention to is that classes which have common students enrolled should not have their exams scheduled at the same time.

Otherwise, if the student rosters of two courses have an empty intersection, then the final exams of these two courses can take place concurrently (and of course it would be desirable to have multiple exams running at the same time in order to have only a few days of final exams).

If we consider a graph with vertex set all the different courses offered in the Fall term, and we assume that there is an edge joining two courses **if and only if their student rosters have a non-empty intersection**, what kind of vertex subsets would we be interested in, that would correspond to groups of courses whose exams can run at the same time?

Answer. We are trying to partition / 'break up' the vertex set (that is, the entire course roster of the Fall term) into   independent sets of vertices (that is, subsets of courses which do not have students in common).

A real-life problem that can be answered via Graph Theory

Scheduling exams: Assume that the Registrar's Office of our university wants to come up with the final exam schedule for the Fall term.

The main restriction that they have to pay attention to is that classes which have common students enrolled should not have their exams scheduled at the same time.

Otherwise, if the student rosters of two courses have an empty intersection, then the final exams of these two courses can take place concurrently (and of course it would be desirable to have multiple exams running at the same time in order to have only a few days of final exams).

If we consider a graph with vertex set all the different courses offered in the Fall term, and we assume that there is an edge joining two courses **if and only if their student rosters have a non-empty intersection**, what kind of vertex subsets would we be interested in, that would correspond to groups of courses whose exams can run at the same time?

Answer. We are trying to partition / 'break up' the vertex set (that is, the entire course roster of the Fall term) into *(as few as possible)* independent sets of vertices (that is, subsets of courses which do not have students in common).

Such partitions will be called 'vertex colourings'

For the following definition, we will be using the set

$$\mathbb{N}_+ = \{1, 2, 3, \ldots\}$$

of positive integers. We will be thinking of each positive integer as a different **colour** (even though most of the time we will keep unspecified which integer corresponds to which colour).

Such partitions will be called 'vertex colourings'

For the following definition, we will be using the set

$$\mathbb{N}_+ = \{1, 2, 3, \dots\}$$

of positive integers. We will be thinking of each positive integer as a different **colour** (even though most of the time we will keep unspecified which integer corresponds to which colour).

Definition

Let $G = (V, E)$ be a graph. A *vertex colouring* of G is any function

$$\xi : V(G) \to \mathbb{N}_+.$$

Such partitions will be called 'vertex colourings'

For the following definition, we will be using the set

$$\mathbb{N}_+ = \{1, 2, 3, \ldots\}$$

of positive integers. We will be thinking of each positive integer as a different **colour** (even though most of the time we will keep unspecified which integer corresponds to which colour).

Definition

Let $G = (V, E)$ be a graph. A *vertex colouring* of G is any function

$$\xi : V(G) \to \mathbb{N}_+.$$

The vertex subsets $V_i = \{v \in V(G) : \xi(v) = i\}$ will be called the *colour classes* of the vertex colouring ξ. Observe that the **non-empty** colour classes form a partition of $V(G)$.

Such partitions will be called 'vertex colourings'

For the following definition, we will be using the set

$$\mathbb{N}_+ = \{1, 2, 3, \ldots\}$$

of positive integers. We will be thinking of each positive integer as a different **colour** (even though most of the time we will keep unspecified which integer corresponds to which colour).

Definition

Let $G = (V, E)$ be a graph. A *vertex colouring* of G is any function

$$\xi : V(G) \to \mathbb{N}_+.$$

The vertex subsets $V_i = \{v \in V(G) : \xi(v) = i\}$ will be called the *colour classes* of the vertex colouring ξ. Observe that the **non-empty** colour classes form a partition of $V(G)$.

— A vertex colouring of G is called a *proper (vertex) colouring* if

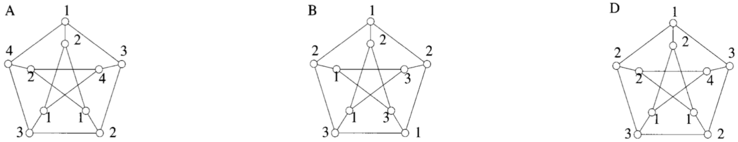
no two adjacent vertices belong to the same colour class.

In other words, if each of the colour classes is an independent set of vertices.

‘Drawing’ vertex colourings

Despite what the term suggests, not every vertex colouring has to be represented by actually colouring the vertices, even though this is always an option if it’s practical (that is, if not many colours appear in the range of the colouring map).

Alternatively, we can write next to each vertex the integer it is mapped to *(see following image containing different vertex colourings of the Petersen graph).*

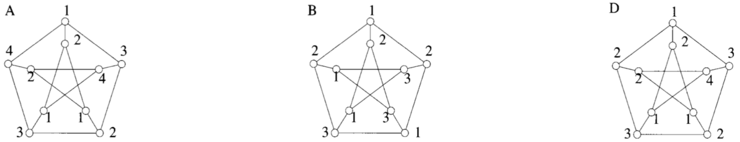


from Wallis’ book

'Drawing' vertex colourings

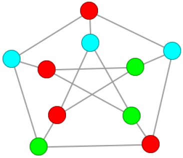
Despite what the term suggests, not every vertex colouring has to be represented by actually colouring the vertices, even though this is always an option if it's practical (that is, if not many colours appear in the range of the colouring map).

Alternatively, we can write next to each vertex the integer it is mapped to *(see following image containing different vertex colourings of the Petersen graph).*



from Wallis' book

Of course, vertex colouring B (for instance) can also be clearly conveyed in this way:



Chromatic number of a graph

From now on we focus almost exclusively on proper colourings of graphs.

Definition 1

Let $G = (V, E)$ be a graph. A proper vertex colouring ξ of G is called an *n-colouring* if there are exactly n non-empty colour classes of ξ. In other words, if the range of the function ξ contains exactly n positive integers.

Chromatic number of a graph

From now on we focus almost exclusively on proper colourings of graphs.

Definition 1

Let $G = (V, E)$ be a graph. A proper vertex colouring ξ of G is called an *n-colouring* if there are exactly n non-empty colour classes of ξ. In other words, if the range of the function ξ contains exactly n positive integers.

G will be called n-colourable if we can find a (proper) n-colouring of G *(pictorially we can think of this as follows: G is n-colourable if n colours are enough for us to find a way to colour the vertices of G so that no two adjacent vertices will end up having the same colour)*.

Chromatic number of a graph

From now on we focus almost exclusively on proper colourings of graphs.

Definition 1

Let $G = (V, E)$ be a graph. A proper vertex colouring ξ of G is called an *n-colouring* if there are exactly n non-empty colour classes of ξ. In other words, if the range of the function ξ contains exactly n positive integers.

G will be called n-colourable if we can find a (proper) n-colouring of G *(pictorially we can think of this as follows: G is n-colourable if n colours are enough for us to find a way to colour the vertices of G so that no two adjacent vertices will end up having the same colour)*.

Definition 2

The *chromatic number* of a graph G is equal to the **smallest** integer n for which we can find a (proper) n-colouring of G.

If n_0 is this smallest integer, then we say that G is *n_0-chromatic*.

Chromatic number of a graph

From now on we focus almost exclusively on proper colourings of graphs.

Definition 1

Let $G = (V, E)$ be a graph. A proper vertex colouring ξ of G is called an *n-colouring* if there are exactly n non-empty colour classes of ξ. In other words, if the range of the function ξ contains exactly n positive integers.

G will be called n-colourable if we can find a (proper) n-colouring of G *(pictorially we can think of this as follows: G is n-colourable if n colours are enough for us to find a way to colour the vertices of G so that no two adjacent vertices will end up having the same colour)*.

Definition 2

The *chromatic number* of a graph G is equal to the **smallest** integer n for which we can find a (proper) n-colouring of G.

If n_0 is this smallest integer, then we say that G is *n_0-chromatic*. We denote this smallest integer by $\chi(G)$ *(in other words, G is $\chi(G)$-chromatic)*.

Chromatic number of a graph

From now on we focus almost exclusively on proper colourings of graphs.

Definition 1

Let $G = (V, E)$ be a graph. A proper vertex colouring ξ of G is called an *n-colouring* if there are exactly n non-empty colour classes of ξ. In other words, if the range of the function ξ contains exactly n positive integers.

G will be called n-colourable if we can find a (proper) n-colouring of G *(pictorially we can think of this as follows: G is n-colourable if n colours are enough for us to find a way to colour the vertices of G so that no two adjacent vertices will end up having the same colour)*.

Definition 2

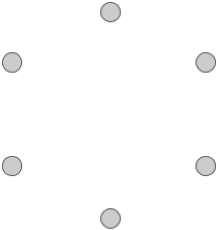
The *chromatic number* of a graph G is equal to the **smallest** integer n for which we can find a (proper) n-colouring of G.

If n_0 is this smallest integer, then we say that G is *n_0-chromatic*. We denote this smallest integer by $\chi(G)$ *(in other words, G is $\chi(G)$-chromatic)*.

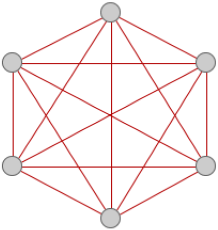
Finally, a $\chi(G)$-colouring of G, that is, a proper colouring of G in $\chi(G)$ colours, is called *minimal*.

Finding the chromatic number of different graphs

We start with basic examples:

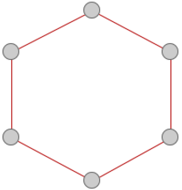


A null graph on 6 vertices

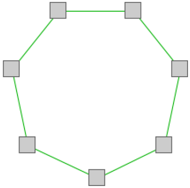


The graph K_6

Finding the chromatic number of different graphs

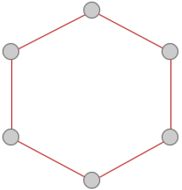


A cycle graph on 6 vertices

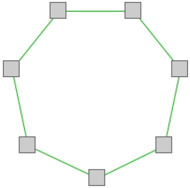


A cycle graph on 7 vertices

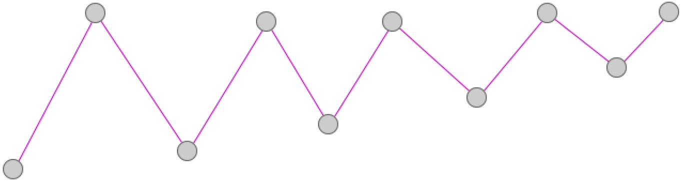
Finding the chromatic number of different graphs



A cycle graph on 6 vertices



A cycle graph on 7 vertices



A path on 10 vertices

Some immediate remarks

- The null graphs are the only graphs which have chromatic number equal to 1 (why?).

Some immediate remarks

- The null graphs are the only graphs which have chromatic number equal to 1 (why?).

- For any graph G of order n, we have that $\chi(G) \leqslant n$.

  Moreover, the complete graph on n vertices is the **only** graph **of order n** which has chromatic number equal to n (why?).

Some immediate remarks

- The null graphs are the only graphs which have chromatic number equal to 1 (why?).
- For any graph G of order n, we have that $\chi(G) \leqslant n$.
  Moreover, the complete graph on n vertices is the **only** graph **of order n** which has chromatic number equal to n (why?).
- Every path (on at least 2 vertices) has chromatic number equal to 2.
  (Practice Question: Could we make a similarly general statement about other trees?

Some immediate remarks

- The null graphs are the only graphs which have chromatic number equal to 1 (why?).

- For any graph G of order n, we have that $\chi(G) \leqslant n$.
  Moreover, the complete graph on n vertices is the **only** graph **of order n** which has chromatic number equal to n (why?).

- Every path (on at least 2 vertices) has chromatic number equal to 2.
  (Practice Question: Could we make a similarly general statement about other trees? We could perhaps use induction on the number n of vertices here in order to approach this question; we'll see another, more direct approach too by the end of this lecture.)

Some immediate remarks

- The null graphs are the only graphs which have chromatic number equal to 1 (why?).

- For any graph G of order n, we have that $\chi(G) \leqslant n$.
  Moreover, the complete graph on n vertices is the **only** graph **of order *n*** which has chromatic number equal to n (why?).

- Every path (on at least 2 vertices) has chromatic number equal to 2.
  (Practice Question: Could we make a similarly general statement about other trees? We could perhaps use induction on the number n of vertices here in order to approach this question; we'll see another, more direct approach too by the end of this lecture.)

- Every **even** cycle has chromatic number equal to 2 (why?).

Some immediate remarks

- The null graphs are the only graphs which have chromatic number equal to 1 (why?).
- For any graph G of order n, we have that $\chi(G) \leqslant n$.
  Moreover, the complete graph on n vertices is the **only** graph **of order *n*** which has chromatic number equal to n (why?).
- Every path (on at least 2 vertices) has chromatic number equal to 2.
  (Practice Question: Could we make a similarly general statement about other trees? We could perhaps use induction on the number n of vertices here in order to approach this question; we'll see another, more direct approach too by the end of this lecture.)
- Every **even** cycle has chromatic number equal to 2 (why?).
- Every **odd** cycle has chromatic number equal to 3 (why?).

Some immediate remarks

- The null graphs are the only graphs which have chromatic number equal to 1 (why?).

- For any graph G of order n, we have that $\chi(G) \leqslant n$.
  Moreover, the complete graph on n vertices is the **only** graph **of order n** which has chromatic number equal to n (why?).

- Every path (on at least 2 vertices) has chromatic number equal to 2.
  (Practice Question: Could we make a similarly general statement about other trees? We could perhaps use induction on the number n of vertices here in order to approach this question; we'll see another, more direct approach too by the end of this lecture.)

- Every **even** cycle has chromatic number equal to 2 (why?).

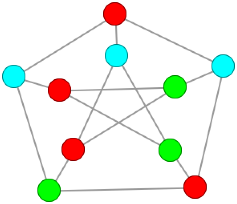
- Every **odd** cycle has chromatic number equal to 3 (why?).

- If G is a graph, and H is a subgraph of G, that is, $H \subseteq G$, then we have that

$$\chi(H) \leqslant \chi(G).$$

Chromatic number of other graphs

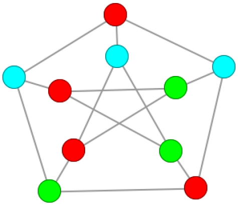
We have already seen that the Petersen graph $\mathcal{P}$ is 3-colourable (this is confirmed by the following colouring):



Observe that this immediately gives us that $\chi(\mathcal{P}) \leqslant 3$.

Chromatic number of other graphs

We have already seen that the Petersen graph $\mathcal{P}$ is 3-colourable (this is confirmed by the following colouring):

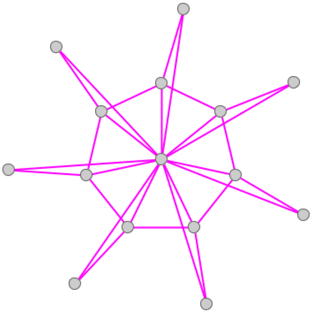


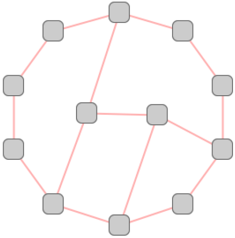
Observe that this immediately gives us that $\chi(\mathcal{P}) \leqslant 3$.

Question. Can you determine precisely $\chi(\mathcal{P})$, and also give a justification for your answer?

Chromatic number of other graphs

Question. Can you determine the chromatic number of the following graphs?





Important Results about the Chromatic Number

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

Important Results about the Chromatic Number

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

Remark. Note that the upper bound given by the theorem is best possible: **if G_0 is a complete graph, or G_0 is an odd cycle**, then

$$\chi(G_0) = \Delta(G_0) + 1.$$

Important Results about the Chromatic Number

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

Remark. Note that the upper bound given by the theorem is best possible: **if G_0 is a complete graph, or G_0 is an odd cycle**, then

$$\chi(G_0) = \Delta(G_0) + 1.$$

However, it turns out these are **essentially** the only examples where the inequality is not strict. This is because of

Important Results about the Chromatic Number

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

Remark. Note that the upper bound given by the theorem is best possible: **if G_0 is a complete graph, or G_0 is an odd cycle**, then

$$\chi(G_0) = \Delta(G_0) + 1.$$

However, it turns out these are **essentially** the only examples where the inequality is not strict. This is because of

Theorem 2 (Brooks' theorem, 1941)

Let G be a connected graph which is neither a complete graph nor an odd cycle. Then $\chi(G) \leqslant \Delta(G)$.

Important Results about the Chromatic Number

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

Remark. Note that the upper bound given by the theorem is best possible: **if G_0 is a complete graph, or G_0 is an odd cycle**, then

$$\chi(G_0) = \Delta(G_0) + 1.$$

However, it turns out these are **essentially** the only examples where the inequality is not strict. This is because of

Theorem 2 (Brooks' theorem, 1941)

Let G be a connected graph which is neither a complete graph nor an odd cycle. Then $\chi(G) \leqslant \Delta(G)$.

We'll give the proof of Theorem 1 shortly.

Subgraphs that require many colours

Definition

Let $G = (V, E)$ be a graph. A *clique* of G is a subgraph H of G which is a complete graph on the vertices it contains.

In other words, a subgraph H of G with vertex set $V(H) \subseteq V = V(G)$ is a clique of G if

- any two vertices in $V(H)$ are joined in G,
- and H is the induced subgraph on those vertices.

Subgraphs that require many colours

Definition

Let $G = (V, E)$ be a graph. A *clique* of G is a subgraph H of G which is a complete graph on the vertices it contains.

In other words, a subgraph H of G with vertex set $V(H) \subseteq V = V(G)$ is a clique of G if

- any two vertices in $V(H)$ are joined in G,
- and H is the induced subgraph on those vertices.

- A clique H of G will be called *maximal* if we cannot view H as the subgraph of a larger clique in G. That is, if we cannot find any vertex u of G outside the vertex set $V(H)$ of H such that the induced subgraph on $V(H) \cup \{u\}$ would again be a clique.

Subgraphs that require many colours

Definition

Let $G = (V, E)$ be a graph. A *clique* of G is a subgraph H of G which is a complete graph on the vertices it contains.

In other words, a subgraph H of G with vertex set $V(H) \subseteq V = V(G)$ is a clique of G if

- any two vertices in $V(H)$ are joined in G,
- and H is the induced subgraph on those vertices.

- A clique H of G will be called *maximal* if we cannot view H as the subgraph of a larger clique in G. That is, if we cannot find any vertex u of G outside the vertex set $V(H)$ of H such that the induced subgraph on $V(H) \cup \{u\}$ would again be a clique.

- A *maximum* clique of G is a clique of G with maximum possible order. The order of a maximum clique in G is called the *clique number* of G, and is denoted by $\omega(G)$.

Subgraphs that require many colours

Definition

Let $G = (V, E)$ be a graph. A *clique* of G is a subgraph H of G which is a complete graph on the vertices it contains.

In other words, a subgraph H of G with vertex set $V(H) \subseteq V = V(G)$ is a clique of G if

- any two vertices in $V(H)$ are joined in G,
- and H is the induced subgraph on those vertices.

- A clique H of G will be called *maximal* if we cannot view H as the subgraph of a larger clique in G. That is, if we cannot find any vertex u of G outside the vertex set $V(H)$ of H such that the induced subgraph on $V(H) \cup \{u\}$ would again be a clique.
- A *maximum* clique of G is a clique of G with maximum possible order. The order of a maximum clique in G is called the *clique number* of G, and is denoted by $\omega(G)$.

Remark. Given a graph G, we can think of the cliques in G and their vertex sets as being the "complementary" concept to the different independent sets of vertices in G.

Subgraphs that require many colours (cont.)

Theorem 3

For every graph G, we have that $\chi(G) \geqslant \omega(G)$.

Subgraphs that require many colours (cont.)

Theorem 3

For every graph G, we have that $\chi(G) \geqslant \omega(G)$.

Proof. By the definition of the clique number $\omega(G)$ of G, we can find a subgraph H_0 of G which is a complete graph and has order $\omega(G)$.

Subgraphs that require many colours (cont.)

Theorem 3

For every graph G, we have that $\chi(G) \geqslant \omega(G)$.

Proof. By the definition of the clique number $\omega(G)$ of G, we can find a subgraph H_0 of G which is a complete graph and has order $\omega(G)$.

Then we know that $\chi(H_0) = \omega(G)$, and we also have $\chi(H_0) \leqslant \chi(G)$. This gives the desired conclusion.

One more result bounding the chromatic number of a graph

Theorem 4

For every graph G of order n, we have that

$$\frac{n}{\alpha(G)} \leqslant \chi(G) \leqslant n + 1 - \alpha(G)$$

where $\alpha(G)$ is the independence number of G, that is, the maximum cardinality of an independent set of vertices in G.

Proof of Theorem 4

- We begin by showing the lower bound $\chi(G) \geqslant \frac{n}{\alpha(G)}$.

Proof of Theorem 4

- We begin by showing the lower bound $\chi(G) \geqslant \frac{n}{\alpha(G)}$.

Consider a $\chi(G)$-colouring ξ_0 of G. Then there are $\chi(G)$ non-empty colour classes of ξ_0, say the classes $V_1, V_2, \ldots, V_{\chi(G)}$, and each of these classes is an independent set of vertices of G.

It follows that the cardinality of each of these classes is $\leqslant$ the maximum cardinality of an independent set in G, or in other words $|V_i| \leqslant \alpha(G)$ for all $i \in \{1, 2, \ldots, \chi(G)\}$.

Proof of Theorem 4

- We begin by showing the lower bound $\chi(G) \geqslant \frac{n}{\alpha(G)}$.

Consider a $\chi(G)$-colouring ξ_0 of G. Then there are $\chi(G)$ non-empty colour classes of ξ_0, say the classes $V_1, V_2, \ldots, V_{\chi(G)}$, and each of these classes is an independent set of vertices of G.

It follows that the cardinality of each of these classes is $\leqslant$ the maximum cardinality of an independent set in G, or in other words $|V_i| \leqslant \alpha(G)$ for all $i \in \{1, 2, \ldots, \chi(G)\}$.

At the same time, the collection $\{V_1, V_2, \ldots, V_{\chi(G)}\}$ is a partition of the vertex set of G (that is, each vertex of G is contained in some colour class, and any two different colour classes have no common vertices). Thus,

$$\begin{aligned} n = |V(G)| &= |V_1| + |V_2| + \cdots + |V_{\chi(G)}| \\ &\leqslant \alpha(G) + \alpha(G) + \cdots + \alpha(G) = \alpha(G) \cdot \chi(G), \end{aligned}$$

or more simply $n \leqslant \alpha(G) \cdot \chi(G)$, which is equivalent to the bound we wanted.

Proof of Theorem 4 (cont.)

- We now show the upper bound $\chi(G) \leqslant n + 1 - \alpha(G)$.

Proof of Theorem 4 (cont.)

- We now show the upper bound $\chi(G) \leqslant n + 1 - \alpha(G)$.

By definition of the independence number $\alpha(G)$ of G, we can find an independent set of vertices V' of G which will contain $\alpha(G)$ vertices. Given that this is an independent set, and hence no two vertices in it are adjacent, we can colour all the vertices in V' using 1 colour.

Proof of Theorem 4 (cont.)

- We now show the upper bound $\chi(G) \leqslant n + 1 - \alpha(G)$.

By definition of the independence number $\alpha(G)$ of G, we can find an independent set of vertices V' of G which will contain $\alpha(G)$ vertices. Given that this is an independent set, and hence no two vertices in it are adjacent, we can colour all the vertices in V' using 1 colour.

Next, we observe that there are $n - \alpha(G)$ vertices in $V(G) \setminus V'$, and each of these vertices will be adjacent to at least one vertex in V' (otherwise V' would not be a maximal independent set).

Proof of Theorem 4 (cont.)

- We now show the upper bound $\chi(G) \leqslant n + 1 - \alpha(G)$.

By definition of the independence number $\alpha(G)$ of G, we can find an independent set of vertices V' of G which will contain $\alpha(G)$ vertices. Given that this is an independent set, and hence no two vertices in it are adjacent, we can colour all the vertices in V' using 1 colour.

Next, we observe that there are $n - \alpha(G)$ vertices in $V(G) \setminus V'$, and each of these vertices will be adjacent to at least one vertex in V' (otherwise V' would not be a maximal independent set).

So the safest option here is to colour the remaining $n - \alpha(G)$ vertices using $n - \alpha(G)$ **new** colours (in other words, each of the vertices in $V(G) \setminus V'$ gets its own colour).

Proof of Theorem 4 (cont.)

- We now show the upper bound $\chi(G) \leqslant n + 1 - \alpha(G)$.

By definition of the independence number $\alpha(G)$ of G, we can find an independent set of vertices V' of G which will contain $\alpha(G)$ vertices. Given that this is an independent set, and hence no two vertices in it are adjacent, we can colour all the vertices in V' using 1 colour.

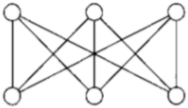
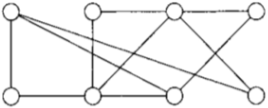
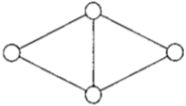
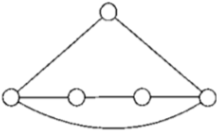
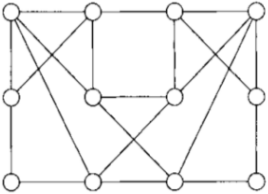
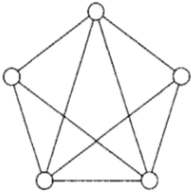
Next, we observe that there are $n - \alpha(G)$ vertices in $V(G) \setminus V'$, and each of these vertices will be adjacent to at least one vertex in V' (otherwise V' would not be a maximal independent set).

So the safest option here is to colour the remaining $n - \alpha(G)$ vertices using $n - \alpha(G)$ **new** colours (in other words, each of the vertices in $V(G) \setminus V'$ gets its own colour).

In the end, we have found a proper colouring of G which uses $(n - \alpha(G)) + 1 = n + 1 - \alpha(G)$ colours. Thus $\chi(G) \leqslant n + 1 - \alpha(G)$.

Vertex colouring in some examples

Practice Exercise. For each of the following graphs, find its chromatic number, and also give a minimal colouring.

from Wallis' book

Back to Theorem 1

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

Back to Theorem 1

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

Proof. This bound will follow from an algorithm that helps us find a proper colouring for any graph G. First we make two observations:

- the colouring algorithm we will describe is a greedy algorithm;

Back to Theorem 1

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

Proof. This bound will follow from an algorithm that helps us find a proper colouring for any graph G. First we make two observations:

- the colouring algorithm we will describe is a greedy algorithm;
- the algorithm does not necessarily give us a minimal colouring of G, so it does not provide a way of determining $\chi(G)$ (it will only give us an upper bound for $\chi(G)$).

Back to Theorem 1

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

Proof. This bound will follow from an algorithm that helps us find a proper colouring for any graph G. First we make two observations:

- the colouring algorithm we will describe is a greedy algorithm;
- the algorithm does not necessarily give us a minimal colouring of G, so it does not provide a way of determining $\chi(G)$ (it will only give us an upper bound for $\chi(G)$).

Suppose that G has order n. We begin by ordering / labelling the vertices of G as $v_1, v_2, \ldots, v_n$.

— Then we colour vertex v_1 using colour 1.

Back to Theorem 1

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

Proof. This bound will follow from an algorithm that helps us find a proper colouring for any graph G. First we make two observations:

- the colouring algorithm we will describe is a greedy algorithm;
- the algorithm does not necessarily give us a minimal colouring of G, so it does not provide a way of determining $\chi(G)$ (it will only give us an upper bound for $\chi(G)$).

Suppose that G has order n. We begin by ordering / labelling the vertices of G as $v_1, v_2, \ldots, v_n$.

— Then we colour vertex v_1 using colour 1.

— Next we examine whether v_2 is a neighbour of v_1 or not: if it is a neighbour, then we colour it using colour 2; if it is not a neighbour, then we colour v_2 using colour 1 again.

Back to Theorem 1

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

Proof. This bound will follow from an algorithm that helps us find a proper colouring for any graph G. First we make two observations:

- the colouring algorithm we will describe is a greedy algorithm;
- the algorithm does not necessarily give us a minimal colouring of G, so it does not provide a way of determining $\chi(G)$ (it will only give us an upper bound for $\chi(G)$).

Suppose that G has order n. We begin by ordering / labelling the vertices of G as $v_1, v_2, \ldots, v_n$.

— Then we colour vertex v_1 using colour 1.

— Next we examine whether v_2 is a neighbour of v_1 or not: if it is a neighbour, then we colour it using colour 2; if it is not a neighbour, then we colour v_2 using colour 1 again.

— We continue like this. If we assume that we have already coloured vertices $v_1, v_2, \ldots, v_i$ such that no two neighbours have the same colour, we then examine how many of the neighbours of v_{i+1} have already been coloured: say, j neighbours of v_{i+1} have been coloured and we have used s colours to do so, say colours $c_1, c_2, \ldots, c_s$; then we colour v_{i+1} using the **first available colour** from the set

$$\{1, 2, \ldots, n\} \setminus \{c_1, c_2, \ldots, c_s\}.$$

Basic colouring algorithm (cont.)

Suppose that G has order n. We begin by ordering / labelling the vertices of G as $v_1, v_2, \ldots, v_n$.

— Then we colour vertex v_1 using colour 1.

— Next we examine whether v_2 is a neighbour of v_1 or not: if it is a neighbour, then we colour it using colour 2; if it is not a neighbour, then we colour v_2 using colour 1 again.

— We continue like this. If we assume that we have already coloured vertices $v_1, v_2, \ldots, v_i$ such that no two neighbours have the same colour, we then examine how many of the neighbours of v_{i+1} have already been coloured: say, j neighbours of v_{i+1} have been coloured and we have used s colours to do so, say colours $c_1, c_2, \ldots, c_s$; then we colour v_{i+1} using the **first available colour** from the set

$$\{1, 2, \ldots, n\} \setminus \{c_1, c_2, \ldots, c_s\}.$$

Basic colouring algorithm (cont.)

Suppose that G has order n. We begin by ordering / labelling the vertices of G as $v_1, v_2, \ldots, v_n$.

— Then we colour vertex v_1 using colour 1.

— Next we examine whether v_2 is a neighbour of v_1 or not: if it is a neighbour, then we colour it using colour 2; if it is not a neighbour, then we colour v_2 using colour 1 again.

— We continue like this. If we assume that we have already coloured vertices $v_1, v_2, \ldots, v_i$ such that no two neighbours have the same colour, we then examine how many of the neighbours of v_{i+1} have already been coloured: say, j neighbours of v_{i+1} have been coloured and we have used s colours to do so, say colours $c_1, c_2, \ldots, c_s$; then we colour v_{i+1} using the **first available colour** from the set

$$\{1, 2, \ldots, n\} \setminus \{c_1, c_2, \ldots, c_s\}.$$

In fact, since we always return to the smallest integer / colour we can use, **and given that, no matter what the vertex v_{i+1} is, its already coloured neighbours cannot be more than all the neighbours of v_{i+1}, which shows that the s colours used for the already coloured neighbours cannot be more than $\deg(v_{i+1}) \leqslant \Delta(G)$,** we can conclude that we will always be able to choose a colour from the set

$$\{1, 2, \ldots, \Delta(G), \Delta(G) + 1\} \setminus \{c_1, c_2, \ldots, c_s\}.$$

(and to systematise the choice here, we might as well choose the **first available colour** from this set).

Back to Theorem 1

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

Back to Theorem 1

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

It should be clear now that, if we apply the *greedy colouring algorithm* we just described for a given graph G, we will find a proper colouring of G which uses at most $\Delta(G) + 1$ colours. Hence, we must have

$$\chi(G) \leqslant \Delta(G) + 1.$$

Back to Theorem 1

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

It should be clear now that, if we apply the *greedy colouring algorithm* we just described for a given graph G, we will find a proper colouring of G which uses at most $\Delta(G) + 1$ colours. Hence, we must have

$$\chi(G) \leqslant \Delta(G) + 1.$$

Important Remark. The ordering of the vertices that we start with is important: different orderings of the vertices can give us proper colourings which use fewer or more colours.

Back to Theorem 1

Theorem 1

For every graph G, we have that $\chi(G) \leqslant \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of G.

It should be clear now that, if we apply the *greedy colouring algorithm* we just described for a given graph G, we will find a proper colouring of G which uses at most $\Delta(G) + 1$ colours. Hence, we must have

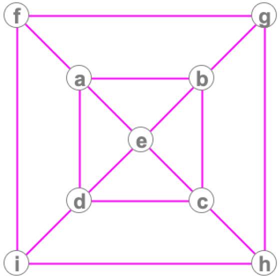
$$\chi(G) \leqslant \Delta(G) + 1.$$

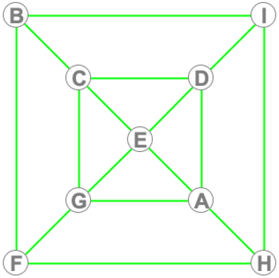
Important Remark. The ordering of the vertices that we start with is important: different orderings of the vertices can give us proper colourings which use fewer or more colours.

In particular, some of the orderings are much preferable than others because they will lead to a minimal colouring. However, figuring out which orderings are optimal is at least as hard as coming up with a minimal colouring.

Applying the algorithm to an example

Apply the greedy colouring algorithm to the following two labellings of the given graph to find proper colourings of the graph (consider an alphabetical ordering of the vertices).





One more result about chromatic numbers

Proposition 1

Let G be a graph that does not contain any odd cycles.

Then $\chi(G) \leqslant 2$. More specifically, $\chi(G) = 1$ if G is a null graph (that is, if it contains no edges), otherwise $\chi(G) = 2$.

One more result about chromatic numbers

Proposition 1

Let G be a graph that does not contain any odd cycles.

Then $\chi(G) \leqslant 2$. More specifically, $\chi(G) = 1$ if G is a null graph (that is, if it contains no edges), otherwise $\chi(G) = 2$.

The proof relies on a result we saw early in the term (in Lecture 8; see page 5 of that slide presentation).

Reminder from Lec 8: Criterion for bipartite graphs

A graph G is a (not necessarily complete) bipartite graph if and only if G does NOT contain any odd cycles.

One more result about chromatic numbers

Proposition 1

Let G be a graph that does not contain any odd cycles.

Then $\chi(G) \leqslant 2$. More specifically, $\chi(G) = 1$ if G is a null graph (that is, if it contains no edges), otherwise $\chi(G) = 2$.

The proof relies on a result we saw early in the term (in Lecture 8; see page 5 of that slide presentation).

Reminder from Lec 8: Criterion for bipartite graphs

A graph G is a (not necessarily complete) bipartite graph if and only if G does NOT contain any odd cycles.

Proof of Proposition 1 (following from this criterion). If G does not contain any odd cycles, then by the criterion we know that we can view G as a subgraph of a complete bipartite graph with, say, partite sets A and B.

One more result about chromatic numbers

Proposition 1

Let G be a graph that does not contain any odd cycles.

Then $\chi(G) \leqslant 2$. More specifically, $\chi(G) = 1$ if G is a null graph (that is, if it contains no edges), otherwise $\chi(G) = 2$.

The proof relies on a result we saw early in the term (in Lecture 8; see page 5 of that slide presentation).

Reminder from Lec 8: Criterion for bipartite graphs

A graph G is a (not necessarily complete) bipartite graph if and only if G does NOT contain any odd cycles.

Proof of Proposition 1 (following from this criterion). If G does not contain any odd cycles, then by the criterion we know that we can view G as a subgraph of a complete bipartite graph with, say, partite sets A and B.

Then, if we colour all vertices of G that end up in the partite set A with colour 1, and similarly we colour all vertices of G that end up in the partite set B with colour 2, we will have a proper colouring of G. This shows that $\chi(G) \leqslant 2$.

One more result about chromatic numbers

Proposition 1

Let G be a graph that does not contain any odd cycles.

Then $\chi(G) \leqslant 2$. More specifically, $\chi(G) = 1$ if G is a null graph (that is, if it contains no edges), otherwise $\chi(G) = 2$.

The proof relies on a result we saw early in the term (in Lecture 8; see page 5 of that slide presentation).

Reminder from Lec 8: Criterion for bipartite graphs

A graph G is a (not necessarily complete) bipartite graph if and only if G does NOT contain any odd cycles.

Proof of Proposition 1 (following from this criterion). If G does not contain any odd cycles, then by the criterion we know that we can view G as a subgraph of a complete bipartite graph with, say, partite sets A and B.

Then, if we colour all vertices of G that end up in the partite set A with colour 1, and similarly we colour all vertices of G that end up in the partite set B with colour 2, we will have a proper colouring of G. This shows that $\chi(G) \leqslant 2$.

Question. Could we now find precisely the chromatic number of any given tree?