Summary of the course · 1MA170

Vilhelm Agdur[1]

[1] vilhelm.agdur@math.uu.se

December 14, 2023

This document gives a summary of the entire course, with keywords highlighted, in colours indicating whether and how it might appear on the exam.

How this document works

The document goes through each lecture in order, noting what we covered in the lecture. In particular, it highlights the key topics of the course, and marks for each how it may appear on the exam.

In particular, the coding works as follows:

1. If it is highlighted like this, that indicates that you are expected to be familiar with the statement to the degree that you could use it to show some other statement or solve an exercise, if the highlighted statement is provided to you.

   You could also be asked to provide a precise statement given a prompt as to what it is about – so for example if you see Dirac's theorem in this document, an exam question might also be "What does Dirac's theorem about the existence of Hamiltonian paths say?". You are not expected to know the proof of the result.

2. If it is highlighted like this, you are expected to not only know the statement in the same sense as in the previous point, but also to have an idea of the proof of the theorem. So you might be asked to fill in a key step of the proof of the statement, write a precise proof given a prompt of what the general outline is, or write an outline of the idea of the proof.

   So if you see Dirac's theorem in this document, an exam question might give you the proof of the result with the step where the maximal length path is turned into a cycle, and you are asked to fill in that step. Or you could be asked to write a proof, given that the drawings for the proof from the lecture notes are given to you. Or you could be asked to draw those figures and explain the broad idea of taking a maximal length path and showing that it can be turned into a cycle, which must be a Hamilton cycle.

3. If it is highlighted like this, you are not expected to recall the exact statement without a prompt, but you are expected to be able to prove it without a reminder of the idea of the proof. So if you see Dirac's theorem, an exam question might look like "State and prove Dirac's theorem about the existence of Hamilton cycles".

4. If it is highlighted like this, you are expected to know the statement of the theorem without any prompt, but not expected to know the proof.

   So if you see Dirac's theorem in this document, an exam question might be "State Dirac's theorem", but you would not be asked about the proof.

5. If it is highlighted like this, you are expected to know the statement of the result, and additionally to have an idea of the proof. So this is the same as this and this together.

6. Finally, if it is highlighted like this, you are expected to know both the theorem and its proof. If you see Dirac's theorem, that means you could see an exam question just ask "State and prove Dirac's theorem".

For definitions, it of course makes no sense to refer to knowing a proof, so we simply highlight definitions like this if you are expected to know and be able to state the definitions, and like this if you are just expected to be able to use the definition and explain the idea of it if given it, but not to be able to state it.

L2: Eulerianity, simple graphs and subgraphs

In our first lecture of the course we started softly, giving the definitions of a multigraph, a walk, a trail, a path, a circuit, and a cycle.

Then we defined what it means for a graph to be connected, and what its connected components are.

Having made all these definitions, we defined an Eulerian trail to be a trail using every edge exactly once, and stated and proved Euler's theorem on Eulerian paths, which characterizes when a graph is Eulerian in terms of the degree of its vertices.

Then, we stated and proved the handshake lemma, which says that

$$2|E| = \sum_{v \in V} d_v.$$

Having done all this, we defined a simple graph[2], and what a graph morphism of simple graphs is, in terms of which we could then define isomorphism of graphs.

Once we knew what it meant for graphs to be isomorphic, we could define an unlabelled graph to be an isomorphism class of graphs.[3]

We ended the lecture by defining what a subgraph, an induced subgraph, an edge-induced subgraph, and a spanning subgraph is.

[2] Which is of course, for most of the course, the only notion of graph we referred to – so generally we end up just calling these *graphs*.

[3] We largely did not end up actually using this concept – other than in a few counting arguments, where we needed to be clear that we were *not* considering unlabelled but labelled graphs.

L3: Common graph families, trees, and Cayley's theorem

We started the lecture with giving definitions of a couple of commonly occuring graph families: the complete graphs, path graphs, cycle graphs, complete bipartite graphs, and complete multipartite graphs.

Then we moved on to the main topic of the lecture: Trees. We started by proving that any tree on n vertices has $n-1$ edges. We then stated, but did not prove,[4] Cayley's formula on the number of labelled trees on n vertices.

Having done this, we proved a characterisation of trees in terms of three properties equivalent to being a tree. Then, we defined the notion of a spanning tree, and proved that all multigraphs have a spanning tree, assuming the axiom of choice.[5]

[4] The proof was deferred until the next lecture, when we were able to prove it as a corollary of a more general result.

[5] In fact, the two statements – existence of spanning trees for arbitrary graphs and the axiom of choice – are equivalent.

L4: Spectral graph theory and the matrix-tree theorem

We started by defining the adjacency matrix of a graph, then we defined what we mean by a directed graph, and used that to define what an incidence matrix of a graph is.

We then proved that the rank of the incidence matrix equals the number of vertices minus the number of connected components. We then defined our final matrix associated to a graph, the Laplacian Q of a graph, and we showed that the Laplacian satisfies $Q = DD^t$.

Then we did a bunch more linear algebra stuff in order to finally arrive at the Kirchhoff matrix-tree theorem. We then used this to give a proof of Cayley's formula, which we had stated in the previous lecture.

L6: Weights, distances, and minimum spanning trees

We started by defining Prim's algorithm. Then we stated and proved that removing an edge from a tree yields a forest of two trees. Then we proved that Prim's algorithm is correct.

Then, we defined Kruskal's algorithm, and proved that it is correct. After this, we defined the graph distance and thus the diameter of a graph. Having done this, we could define Dijkstra's algorithm.

L7: The max-flow min-cut and marriage theorems

We defined what a weighted directed graph is, and what a flow network is. Then we defined what a flow on these networks is, and its value.

We then defined a cut on a flow network and its capacity, and showed that the value of any flow is upper bounded by the capacity of any cut. We then defined the residual network of a flow, and defined an augmenting path in this network.

We stated that any augmenting path can be used to find a higher-value flow, and used this to show the Ford-Fulkerson theorem.

Then, we defined what a matching and a bipartite graph is, and stated and proved Hall's marriage theorem using the max-flow-min-cut duality we had just seen.

L8: Vertex covers, Hamilton cycles, independent sets

L10: Connectivity

L11: Planarity

L12: Vertex colourings

L14: The probabilistic method and the Erdős-Rényi random graph

L16: Edge-colourings and Ramsey theory

L17: The Rado graph

L18: Extremal graphs and Szemerédi's regularity lemma

L19: ???