1. Aufgabe - 25 Punkte

Man hat 5 Graphen und 6 Eigenschaften und muss für jeden Graph angeben welche Eigenschaften er erfüllt.

Man muss mit 0 (false) und 1 (true) angeben welche Eigenschaft erfüllt bzw. nicht erfüllt ist.

	G1	G2	G3	G4	G5
Eulersch					
Planar					
Es gibt ein perfektes Matching					
Chromatische Zahl = 3?					
Bipartit					
Enthält Teilgraph, der zu $K_{3,3}$ homöomorph ist					

6 richtig angekreuzte Eigenschaften pro Graph bringen 5 Punkte, es gibt 5 Graphen, also 5 * 5 = 25 Punkte

2. Aufgabe - 18 Punkte

Wieder eine Implikationskette verwenden, um 3 Eigenschaften zu beweisen - es ging um Bäumeneigenschaften und zwar - Kreisfreiheit und Zusammenhängend-"keit". Ähnliche Aufgabe ist Aufgabe 30, Übungsblatt 10, SoSe20.

3. Aufgabe - 19 Punkte

Knotenfärbung, du hast den Greedy Algorithmus aus der VL gegeben und musst einige Fragen beantworten - wie viele Farben er benutzt und beweisen, dass es eine Anordnung von Knoten gibt, sodass wenn der Algorithmus die Knoten durchgeht, er so viele Farben benutzen wie die chromatische Zahl. Es gab auch einen bipartiten Graphen, wobei die Knotenmenge war $\{a_1,b_1,a_2,b_2,\dots,a_k,b_k\}$ und jeder Knoten v_i war mit allen Knoten von der anderen Menge verbunden, aber nur ohne diese mit demselben Index. Z.B a_1 ist mit alle b's verbunden ohne b_1. Die Frage war wie viele Kanten gibt es in diesem Graphen und wie groß ist die chromatische Zahl. Man sollte auch sagen, wie viele Farben der Algorithmus bei diesem Graphen benutzen wird.

4. Aufgabe - 20 Punkte

Kantenfärbung, du musst in einen 3-regulären Graph die Kanten färben. Dann musst du beweisen, dass falls ein 3-regulärer Graph einen Hamiltonkreis enthält, so ist sein chromatischer Index 3. Man musste auch beweisen, dass für jeden 3-regulären Graphen die Anzahl der Knoten gerade ist.

5. Aufgabe - 18 Punkte

Für jedes n aus N gibt es folgende Definition: ([n], {{…}, {…}}), wobei n die Anzahl von Knoten in jedem Graph ist und in {…} ist ein gerichteter Pfad gegeben.

Z.B. für n=3:

([3], {{1,2}})

([3], {{1,3}})

([3], {{2,3}})

([3], {{1,2}, {2, 3}})

([3], {{1,2}, {1, 3}})

([3], {{2,3}, {1, 3}})

([3], {{1,2}, {2,3}, {1,3}})

Mit allen diesen Möglichkeiten, Knoten zu verbinden, muss man ein Hasse-Diagramm erstellen.

Man soll auch die Anzahl der Teilmengen von G_n angeben.

Dann muss man eine partielle Ordnung als gerichteten Graph darstellen und auch angeben wie viele Knoten die längste Kette in diesem Graph enthält.

Partielle Ordnungen, du musst für eine Definition das Hasse-Diagramm angeben und dann die partielle Ordnung als gerichteten Graph angeben