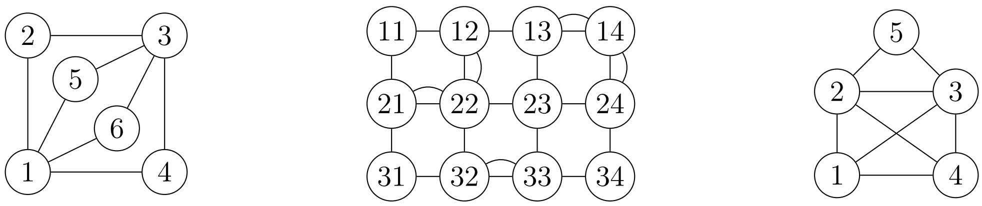
Note: Because there is no proseminar on 8 December, you have two weeks time to work on this sheet. There are also three special bonus exercises. These are *not* associated to this one particular sheet; rather their points get added on top of whatever your point number is at the end of the semester. Note that you have time until after the Christmas break for these.

Exercise 1



G_1 G_2 G_3

a) For each of the (multi-)graphs G_1 and G_2:

   i) Verify that it satisfies the degree condition for being Eulerian.

   ii) Decompose it into disjoint cycles via the method sketched in the lecture.

   iii) Compute an Euler cycle from your cycle decomposition via the method from the lecture.

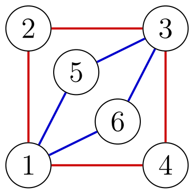
b) Prove: if a multi-graph G has an Euler trail then G has either 0 or 2 nodes with odd degree.

c) Prove: if a connected multi-graph G has 0 or 2 nodes of odd degree, G has an Euler trail.

d) Apply your proof from c) to obtain an Euler trail for G_3 (the *Haus des Nikolaus*).

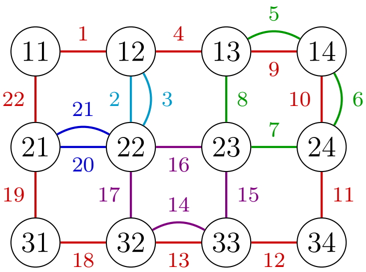
Solution: There are usually many ways to partition an Eulerian graph into cycles. If your solution is different from the one shown here, it is not necessarily wrong.

a) For G_1, we start arbitrarily with the edge $\{1, 2\}$ and extend it to the cycle 1—2—3—4—1. After removing this cycle, we arbitrarily start with the edge $\{3, 6\}$ and obtain the cycle 3—6—1—5—3.



We can join these two cycles e.g. at 3 to obtain the Euler cycle 1—2—3—6—1—5—3—4—1.

For G_2, we can partition the cycles and obtain an Euler cycle as follows. The different colours each show one cycle, and the numbers next to each edge shows its position in the Euler cycle.

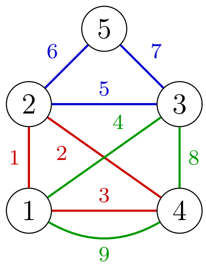


b) Suppose $G = (V, E)$ has an Euler trail p starting at u and ending at v.

- If $u = v$, then p is an Euler cycle and there are 0 nodes with odd degree according to the criterion from the lecture.
- If $u \neq v$, then let $e \notin E$ be a new edge connecting u and v. Now pe is an Euler cycle in the graph $G' = (V, E \cup \{e\})$. Thus all nodes have even degree in G' via the criterion from the lecture. And then u and v have odd degree in G and all others have even degree.

c) Suppose $G = (V, E)$ has 0 or 2 nodes with odd degree. For 0 nodes, the result is immediate since we have an Euler cycle (which is also an Euler trail). For 2 nodes u and v of odd degree, we again let $e \notin E$ be a new edge connecting u and v and let $G' = (V, E \cup \{e\})$. Since all nodes have even degree in G', there exists an Euler cycle p in G'. We can shift this cycle such that it ends with the edge e. Removing e from p yields an Euler trail in G.

d) There are two nodes with odd degree, namely 1 and 4. We add a new edge between them, decompose the resulting graph into three cycles as shown below, and construct an Euler cycle by joining those three cycles.



Deleting the extra edge (9) we added from the circuit yields an Euler trail starting at 1 and ending at 4 in the original graph G_3, namely 1—2—4—3—2—5—3—4.

Exercise 2

Consider the following graphs that were introduced in the lecture: L_n, C_k, K_n, $K_{m,n}$, Q_n, Petersen graph. For each of these graphs, do the following (under the assumptions $m, n \geq 1$ and $k \geq 3$):

a) Find how many nodes and edges it has.

b) Find the min and max degree δ and Δ.

c) Determine if it is k-regular for any k.

d) Find the clique number α and the independence number I.

e) Find the diameter d.

f) Find the chromatic number χ and sketch a node colouring of G with χ colours.

You need not prove your results.

Solution:

- L_1 and K_1 consist of just one node and no edges and thus $\Delta = \delta = d = 0$ and $\alpha = I = 1$. It is a 0-regular graph. The colouring that assigns colour 1 to the node is a valid colouring, thus the graph is 1-colourable and $\chi = 1$.
- L_2, K_2, and $K_{1,1}$ are all the same graph (up to isomorphism) consisting of two nodes and one edge connecting them. We thus have $\Delta = \delta = d = I = 1$ and $\alpha = 2$. It is a 1-regular graph and $\chi = 2$.
- C_k has k nodes and k edges. $\delta(C_k) = \Delta(C_k) = 2$ and thus C_k is 2-regular. The diameter is $\lfloor \frac{k}{2} \rfloor$. The clique number α is 2 (unless $k = 3$ in which case $\alpha = 3$ since C_3 is isomorphic to K_3). The independence number I is $\lfloor \frac{k}{2} \rfloor$ (we obtain the biggest independent set by picking every other node). The chromatic number χ is 2 if k is even (then we can alternate between the colours 1 and 2) and 3 if k is odd (then we can alternate between the colours 1 and 2 but need colour 3 for the last node).
- If $m, n \geq 1$ and $(m, n) \neq (1, 1)$, the $K_{m,n}$ has $m + n$ nodes and $m \cdot n$ edges. $\delta(K_{m,n}) = \min(m, n)$ and $\Delta(K_{m,n}) = \max(m, n)$, so $K_{m,n}$ is k-regular if and only if $m = n = k$. The clique number α is 2 and the independence number I is $\max(m, n)$ (the left and right set are both independent). The diameter is 2, since at least one of the two 'components' has 2 or more nodes, and we need two steps to reach one from the other. The chromatic number is 2 (achieved by assigning colour 1 to the nodes on the left and 2 to the nodes on the right). Thus the complete bipartite graph $K_{m,n}$ is, in fact, bipartite.
- L_n for $n \geq 3$ has n nodes and $n - 1$ edges. $\delta(L_n) = 1$ and $\Delta(L_n) = 2$ and L_n is thus not k-regular for any k. The diameter d is $n - 1$ since the nodes 1 and n have distance $n - 1$. The clique number α is 2 and the independence number I is $\lceil \frac{n}{2} \rceil$ (obtained by picking every other node). The chromatic number χ is 2 since we can alternate between the colours 1 and 2.
- K_n for $n \geq 3$ has n nodes and $\frac{1}{2}n(n+1)$ edges. $\delta(K_n) = \Delta(K_n) = n-1$, so K_n is $n-1$-regular. The diameter is 1 since every node is reachable from every other node by one step. The clique number α is n basically by definition; the independence number I is 1. The chromatic number is n since every node must have a different colour.
- The Petersen graph has 10 nodes and 15 edges and $\delta = \Delta = 3$, so it is 3-regular. It has chromatic number 3, as shown by the 3-colouring from the lecture and the fact that it contains a cycle of odd length and thus cannot have chromatic number less than 3. The clique number is 2 and the independence number 4 (this requires a bit of experimenting, but you can exploit the symmetry of the graph to reduce the number of cases). The diameter is 2.
- Q_n has 2^n nodes. Each node is connected to the nodes that differ from it in exactly one coordinate, i.e. every node has n neighbours and thus $\delta(Q_n) = \Delta(Q_n) = n$ and Q_n is n-regular. The sum of degrees is thus $n \cdot 2^n$, so there are $n \cdot 2^{n-1}$ edges.

  The diameter is n since if we want to reach node $y = (y_1, \ldots, y_n)$ from node $x = (x_1, \ldots, x_n)$ the number of steps is exactly the number of components in which x and y differ – at most n if they all differ. The chromatic number is 2, achieved by assigning colour 1 to all nodes with an even number of zeroes and 2 to all nodes with an odd number of zeroes.

  The clique number α is 2, since if x and y differ in one coordinate and y and z differ in one coordinate then x and z must either be equal or differ in two coordinates (and thus cannot be adjacent).

  The independence number I is equal ot 2^{n-1} (exactly half the nodes) because e.g. all the nodes with an even number of zeroes are an independent set, since any node adjacent to one of them has an odd number of zeroes. We cannot have more than that because whenever we add a node $(x_1, x_2, \ldots, x_n)$ to our independent set, the node $(1 - x_i, x_2, \ldots, x_n)$ is 'blocked', so we can never half more than half of the nodes in an independent set.

Exercise 3
Show that for any $n \geq 5$ there is a connected 4-regular graph with n nodes and no loops.

Hint: Use mathematical induction on n. How can you modify a connected 4-regular graph with n nodes to get one for $n + 1$ nodes?

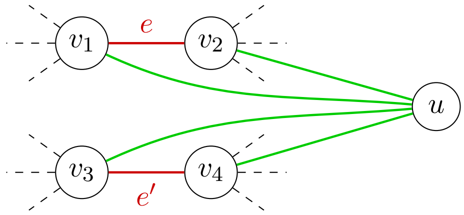
Solution: Following the hint, we show the statement by induction over n.

For the base case $n = 5$, the K_5 is a 4-regular connected graph with n nodes and no loops.

For the induction step, let $n \geq 5$ be arbitrary and assume as induction hypothesis that there exists a connected 4-regular graph $G = (V, E)$ with $|V| = n$ and no loops.

Pick some edge $e = \{v_1, v_2\} \in E$. Now pick some node $v_3 \in V \setminus e$. Because G is 4-regular, v_3 has at least one edge $e' = \{v_3, v_4\}$ such that $v_4 \notin \{v_1, v_2\}$.

Now let $u \notin V$ be a new node and define $G' = (V \cup \{u\}, E \setminus \{e, e'\} \cup \{\{u, v_1\}, \{u, v_2\}, \{u, v_3\}, \{u, v_4\}\})$. In other words: we delete the edges e and e' and instead connect u with the nodes v_1 to v_4.



It is easy to see that G' is then a 4-regular graph with $n + 1$ nodes and no loops. It is also still connected:

- If $x, y \in V$ are nodes in G there must be a path in G between them because G was connected. If that path contains e.g. e, we replace it with the two edges $\{v_1, u\}$ and $\{u, v_2\}$ on the path (and similarly for e'). So x and y are still connected in G'.
- Any node $x \in V$ is also connected to the new node v in G': as we have just seen, there exists a path in G' between x and e.g. v_1. Adding the edge $\{v_1, u\}$ to this path yields a path between x and u.

Note that in order for the argument to work it is crucial that v_1, v_2, v_3, v_4 really are distinct – otherwise the degrees would be less than 4 after this step because we remove two edges from some v_i but only add one edge from u to v_i (since we have a graph, not a multigraph).

Important: If you solve one of these special bonus exercises, do not cross it in OLAT but rather you have to upload your solution in the respective section in OLAT.

Note that plagiarism (i.e. copying significant portions of code from somebody else or some source on the internet) is not allowed and may be penalised. Asking for advice, looking up small problems on StackOverflow etc. is fine – but stay reasonable.

The deadline for these special bonus exercises is 12.01.2023.

Christmas bonus exercise 1 (1 point)
Implement Kruskal's algorithm (the version that computes a minimal spanning forest of a weighted undirected graph). The choice of the proramming language, representation of graphs, data structures is up to you.

Note: For Haskell and Python, you can find templates (including a suitable graph representation and some example input) on OLAT. There is also a file `Partition.hs` (resp. `partition.py`) that provides the implementation of a simple union–find data structure that you can use.

Solution: Reasonably efficient solutions in Haskell and Python are provided in OLAT.

Christmas bonus exercise 2 (1 point)
Implement topological sorting of a dag (try to be efficient!). The choice of the proramming language, representation of graphs, data structures is up to you.

Note: For Haskell and Python, you can find templates (including a suitable graph representation and some example input) on OLAT.

Solution: Reasonably efficient solutions in Haskell and Python are provided in OLAT.

Christmas bonus exercise 3 (2 points)
Implement the algorithms from the lecture that take an Eulerian (multi-)graph and compute a cycle decomposition and, from that an Euler cycle.

Apply your program to some examples from the lecture or this sheet.

You may again choose a programming language of your choice and represent multi-graphs in whatever way you choose. A template is not provided, but feel free to adapt one from the previous exercises.

Solution: A solution in Python is provided in OLAT.