Algorithms and Probability

Exercise Session 1

Me

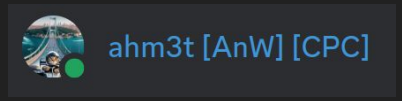
Ahmet Ala

6th semester

1x TA in AnD

ahmala@ethz.ch

discord: ahmetala



Me

6th semester

1x TA in AnD

ahmala@ethz.ch

discord: ahmetala  ahm3t [AnW] [CPC]

Bonus: President of Competitive Programming Committee at VIS

ETH team dETHroners wins SWERC 2024

Over the final weekend of January, two teams representing ETH Zurich took part in the ICPC Southwestern Europe Regional Contest (SWERC) held in Paris. Team dETHroners seized the SWERC title, clinching the gold medal with an impressive 11 problems solved. Congratulations!

06.02.2024

How To Study

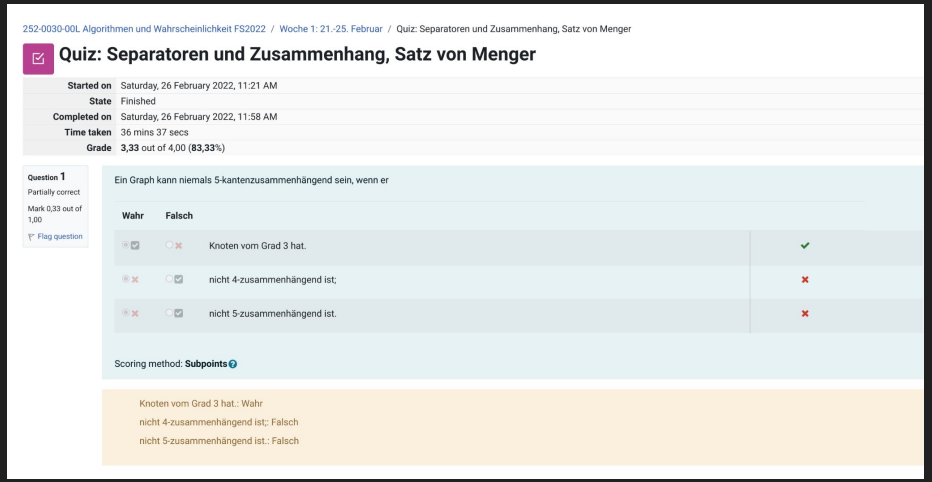
Do the homeworks(Surprise surprise “:D)

Quizzes on Moodle(do not give any bonus)

How To Study

Do the homeworks(Surprise surprise “:D

Quizzes on Moodle(do not give any bonu



How To Study

Do the homeworks(Surprise surprise “:D)

Quizzes on Moodle(do not give any bonus)

Official Script

In case you have problems with German -> PVW Script

Ask questions to me

Ask questions on Moodle

How To Study

Do the homeworks(Surprise s

Quizzes on Moodle(do not giv

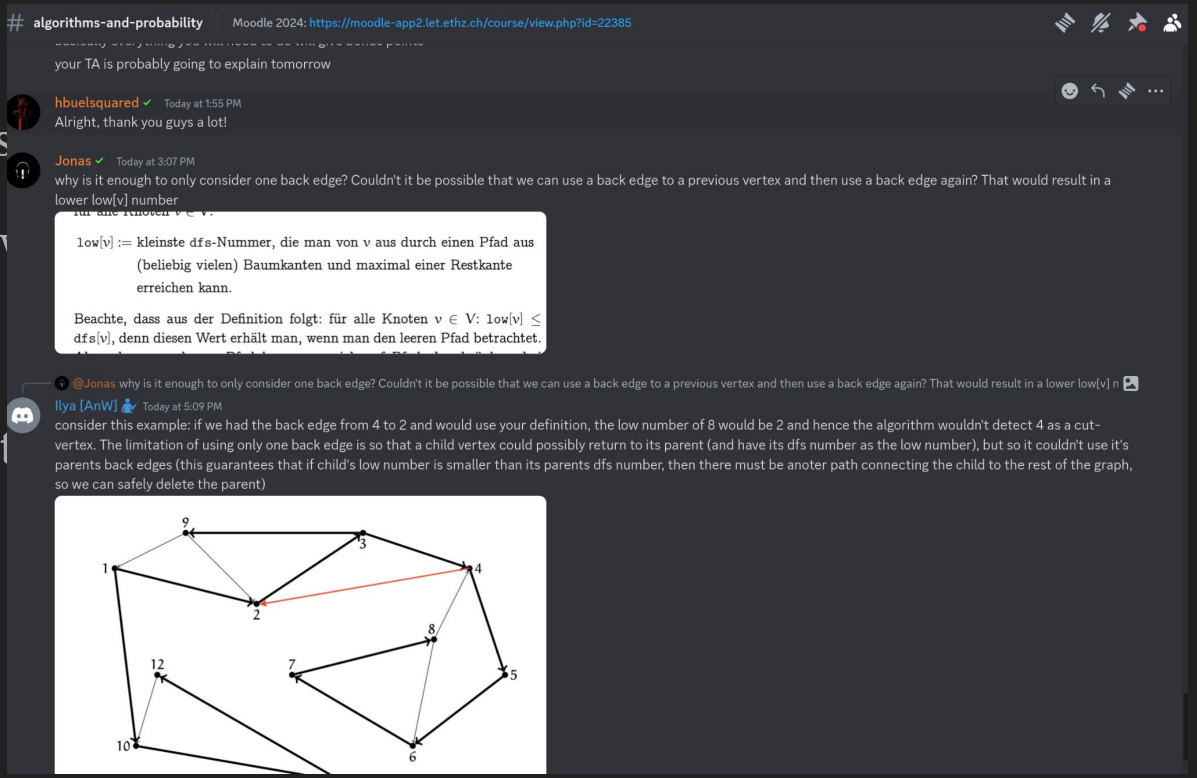
Official Script

In case you have problems wit

Ask questions to me

Ask questions on Moodle

Ask questions on [Discord]



How To Study

Do the homeworks(Surprise surprise “:D)

Quizzes on Moodle(do not give any bonus)

Official Script

In case you have problems with German -> PVW Script

Ask questions to me

Ask questions on Moodle

Ask questions on Discord

Come to the lectures even if you are lost(topics change)

Bonus up to 0.25

Theory: 6*2 points = 12 points

Code Expert: 12*2 points = 24 points

Minitest: 6*2 points = 12 points

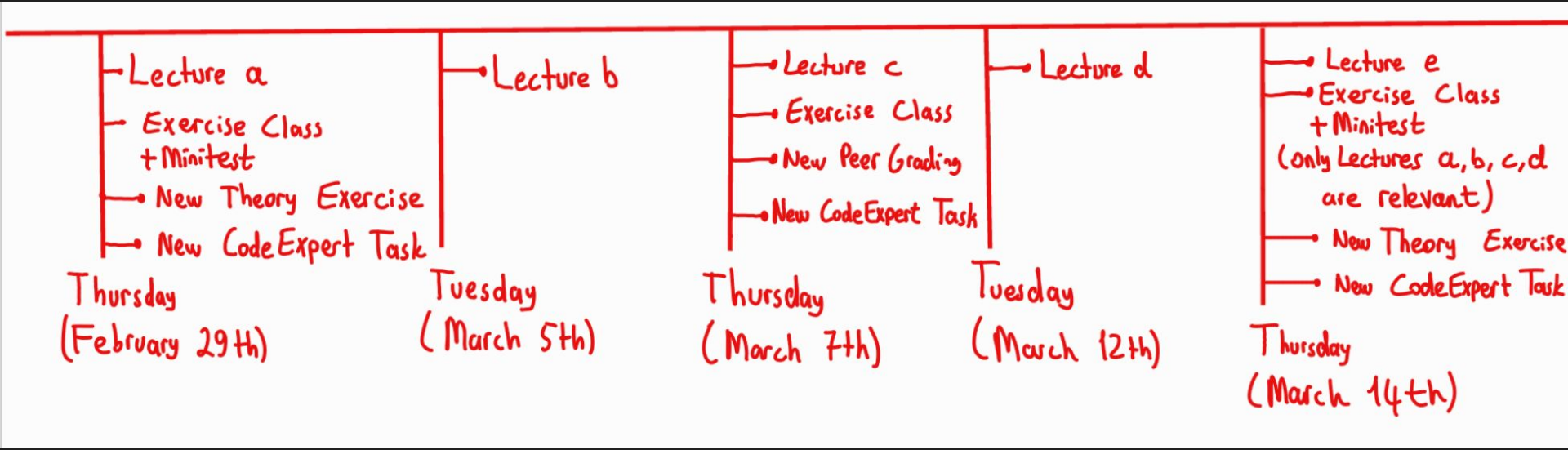
Peer Grading: 5*2 points = 10 points

In Total: 58 points

Bonus up to 0.25

Theory: 6*2 points = 12 points	21%
Code Expert: 12*2 points = 24 points	41%
Minitest: 6*2 points = 12 points	21%
Peer Grading: 5*2 points = 10 points	17%
In Total: 58 points	In Total: 58 points

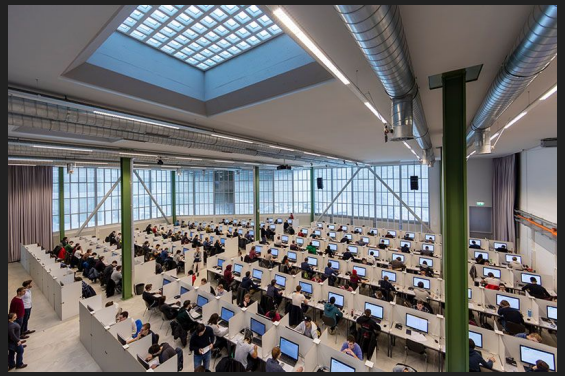
Schedule


Lecture a
Exercise Class + Minitest
New Theory Exercise
New CodeExpert Task
Thursday (February 29th)
Lecture b
Tuesday (March 5th)
Lecture c
Exercise Class
New Peer Grading
New CodeExpert Task
Thursday (March 7th)
Lecture d
Tuesday (March 12th)
Lecture e
Exercise Class + Minitest
(only Lectures a, b, c, d are relevant)
New Theory Exercise
New CodeExpert Task
Thursday (March 14th)

Exam

On the computer(in ONA, HG G1 etc.)

180 minutes

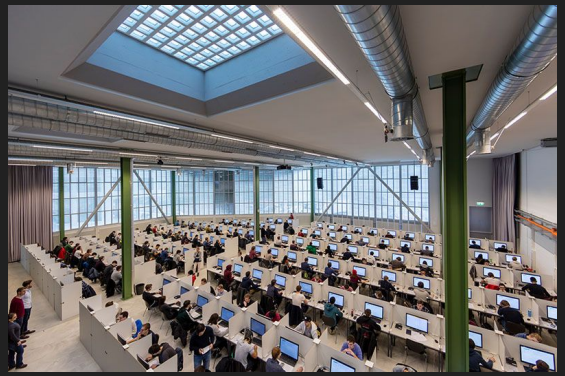


Exam

On the computer(in ONA, HG G1 etc.)

180 minutes

Programming + Theory in the same time

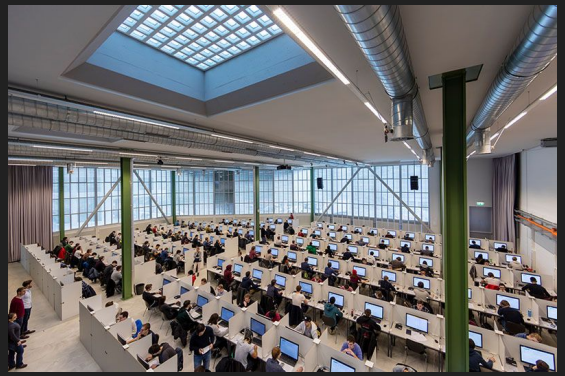


Exam

On the computer(in ONA, HG G1 etc.)

180 minutes

Programming + Theory in the same time



Hard to manage the time, really hard

Exam

On the computer(in ONA, HG G1 etc.)

180 minutes

Programming + Theory in the same time Hard to manage the time, really hard

Following may change, no guarantees:

%50 Short Questions(True/False, MC, Fill in the gaps) Very similar to Minitests

%25 Theory Task - Proving

%25 Programming

Exam

On the computer(in ONA, HG G1 etc.)

180 minutes

Programming + Theory in the same time Hard to manage the time, really hard

Following may change, no guarantees:

%50 Short Questions(True/False, MC, Fill in the gaps) Very similar to Minitests

%25 Theory Task - Proving Very similar to Theory Sheets

%25 Programming

Exam

On the computer(in ONA, HG G1 etc.)

180 minutes

Programming + Theory in the same time Hard to manage the time, really hard

Following may change, no guarantees:

%50 Short Questions(True/False, MC, Fill in the gaps) Very similar to Minitests

%25 Theory Task - Proving Very similar to Theory Sheets

%25 Programming 1 Probability DP Task + 1 MaxFlow Task

Exam

On the computer(in ONA, HG G1 etc.) 3 pages of Formelsammlung

180 minutes

Programming + Theory in the same time Hard to manage the time, really hard

Following may change, no guarantees:

%50 Short Questions(True/False, MC, Fill in the gaps) Very similar to Minitests

%25 Theory Task - Proving Very similar to Theory Sheets

%25 Programming 1 Probability DP Task + 1 MaxFlow Task

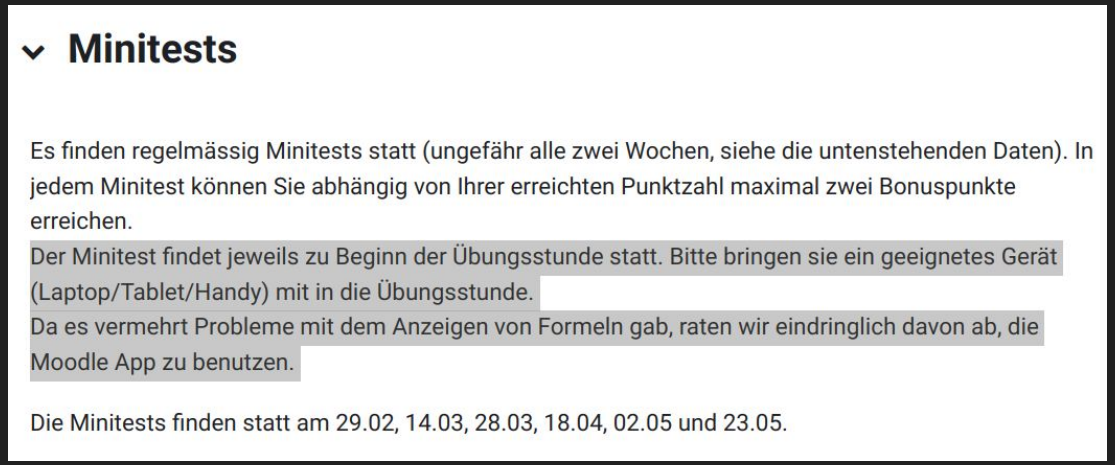
Minitest Next Week

First three lectures are relevant

Minitest Next Week

First three lectures are relevant

Bring Laptop OR use Tablet with Web Browser, not the Moodle App



Minitests

Es finden regelmässig Minitests statt (ungefähr alle zwei Wochen, siehe die untenstehenden Daten). In jedem Minitest können Sie abhängig von Ihrer erreichten Punktzahl maximal zwei Bonuspunkte erreichen.
Der Minitest findet jeweils zu Beginn der Übungsstunde statt. Bitte bringen sie ein geeignetes Gerät (Laptop/Tablet/Handy) mit in die Übungsstunde.
Da es vermehrt Probleme mit dem Anzeigen von Formeln gab, raten wir eindringlich davon ab, die Moodle App zu benutzen.

Die Minitests finden statt am 29.02, 14.03, 28.03, 18.04, 02.05 und 23.05.

k-connected(k-zusammenhängend)

$|V| \geq k+1$

$\forall X \subseteq V, |X| < k : G[V \setminus X]$ is connected

k-connected(k-zusammenhängend)

$|V| \geq k+1$

$\forall X \subseteq V, |X| < k : G[V \setminus X]$ is connected

You must delete at least k-vertices to make the graph disconnected

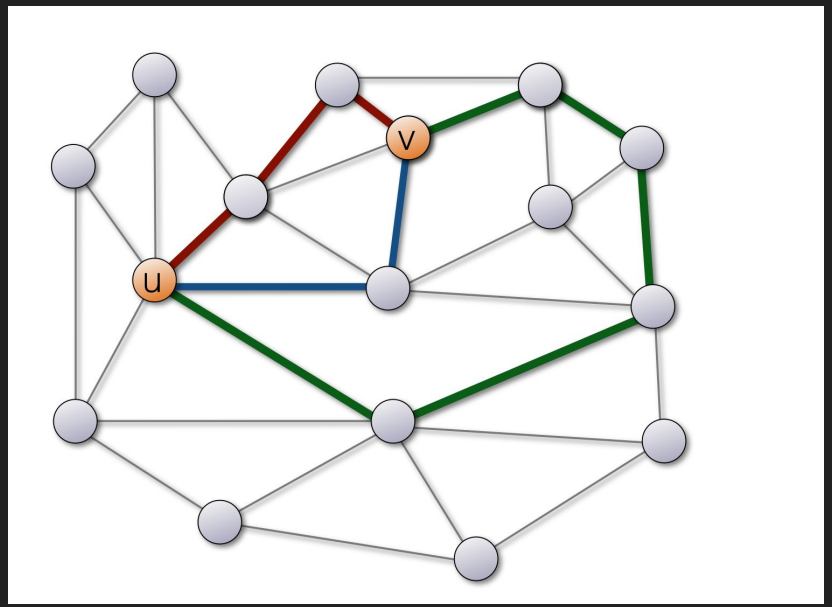
k-edges-connected (k-kanten-zusammenhängend)

$|E| \geq k + 1$

$\forall X \subseteq E, |X| < k : (V, E \setminus X)$ is connected

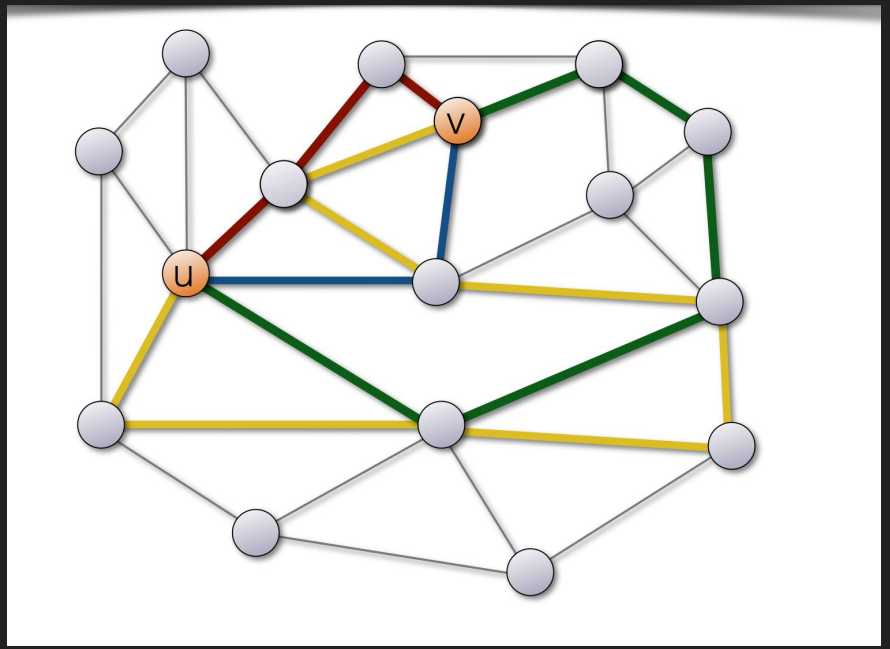
Theorem(Menger)

A graph is k-connected if and only if for all $u, v \in V$, $u \neq v$ there exist at least k internally-vertex-disjoint u-v-paths. (only u, v may appear on multiple paths)



Theorem(Menger)

A graph is k-edge-connected if and only if for all $u, v \in V$, $u \neq v$ there exist at least k edge-disjoint u-v-paths. (no edges may appear on multiple paths)



There exists a vertex with degree < k

Is the graph k-connected?

Connected <= edge-connected <= minimum degree

Correct or Not?

Edge-disjoint-u-v-path -> vertex-disjoint-u-v path

Correct

K-connected -> k-edge-connected

Not Correct

K-connected <- k-edge-connected

Cut vertex = Articulation point = Artikulationsknote

articulation point is a node whose removal increases the number of connected components in the graph

Cut edge = Bridge = Brücke

bridge is an edge whose removal increases the number of connected components in the graph

Correct or Not?

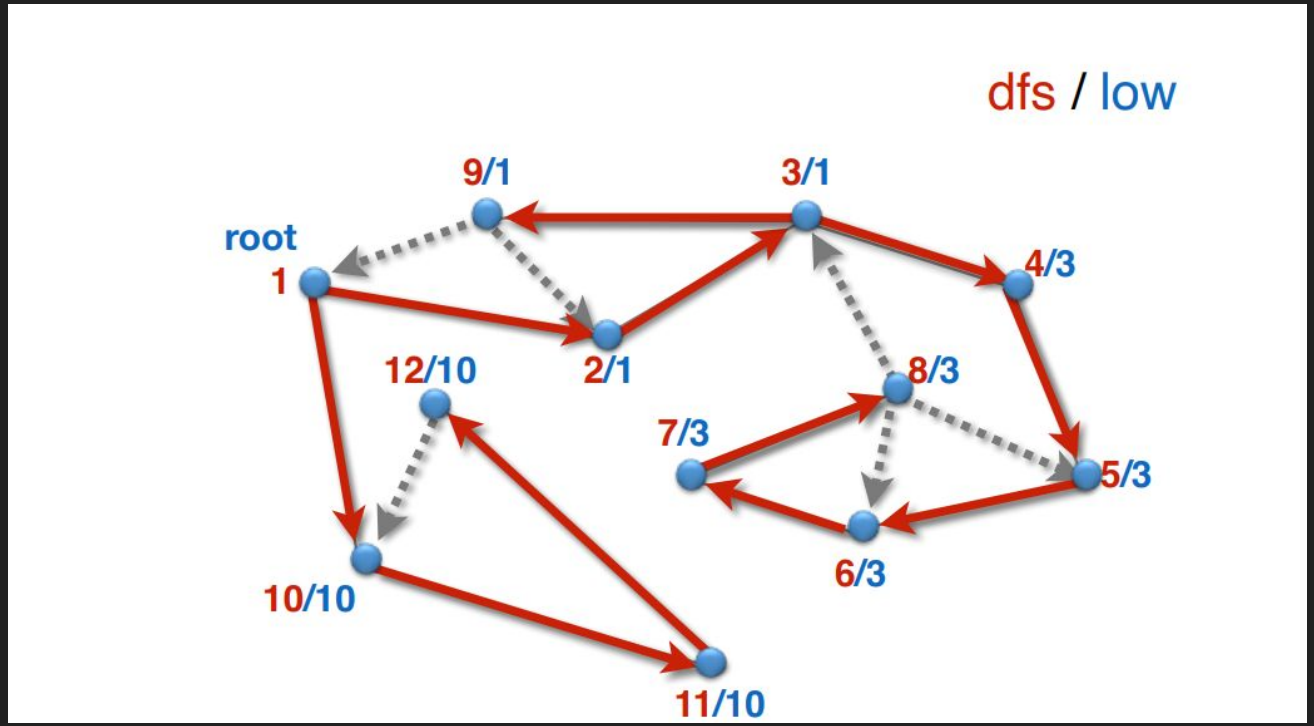
Each articulation point is incident to at least one bridge

Correct or Not?

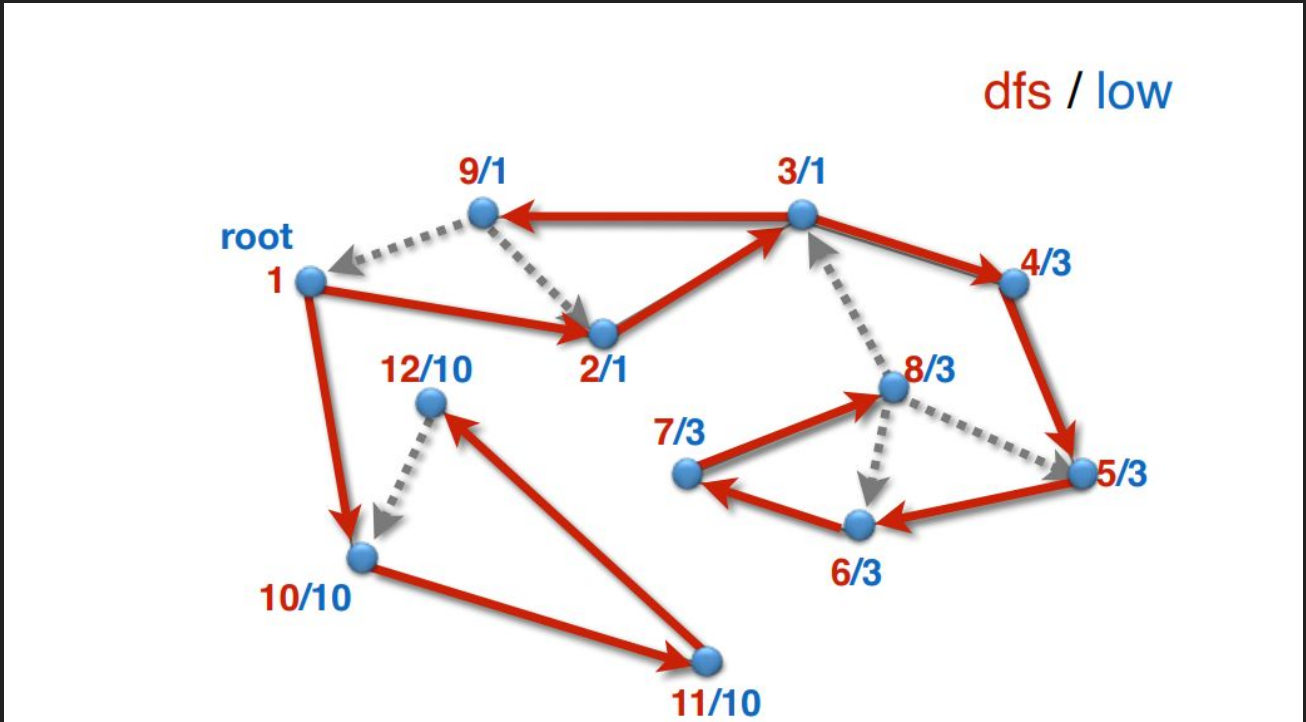
Each bridge has one articulation point as an end-point

DFS-LOW Values

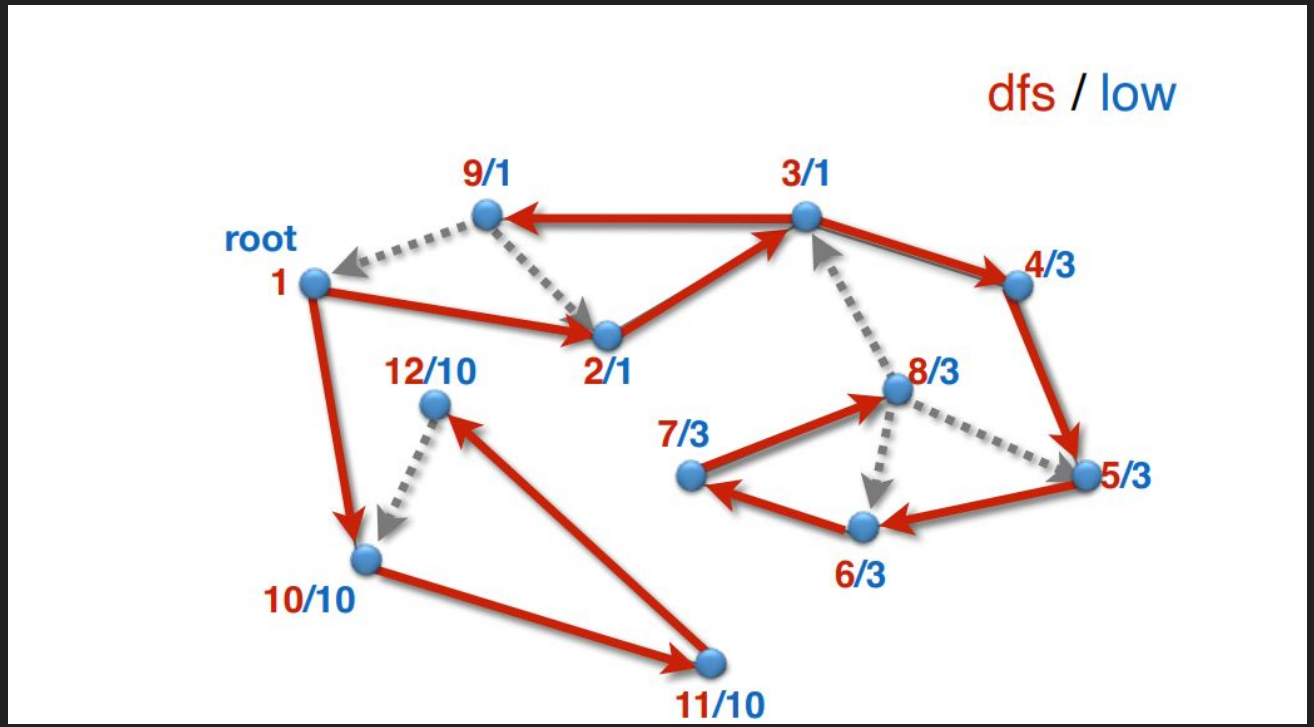
$low[v]$ = minimum dfs number of all vertices reachable from v via arbitrarily many edges in the search tree and at most one other edge that is in the graph


dfs / low
9/1
3/1
root
1
4/3
2/1
12/10
8/3
7/3
5/3
6/3
10/10
11/10

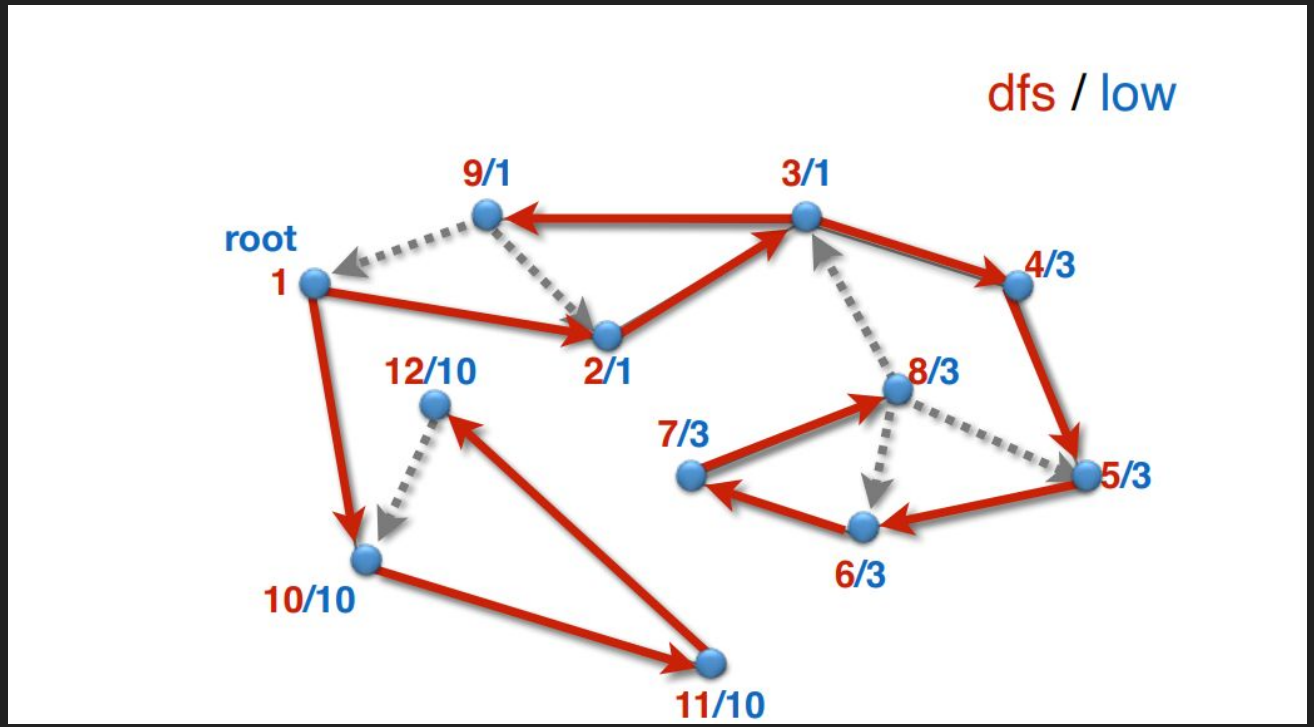
Who are the kids of root in dfs tree?



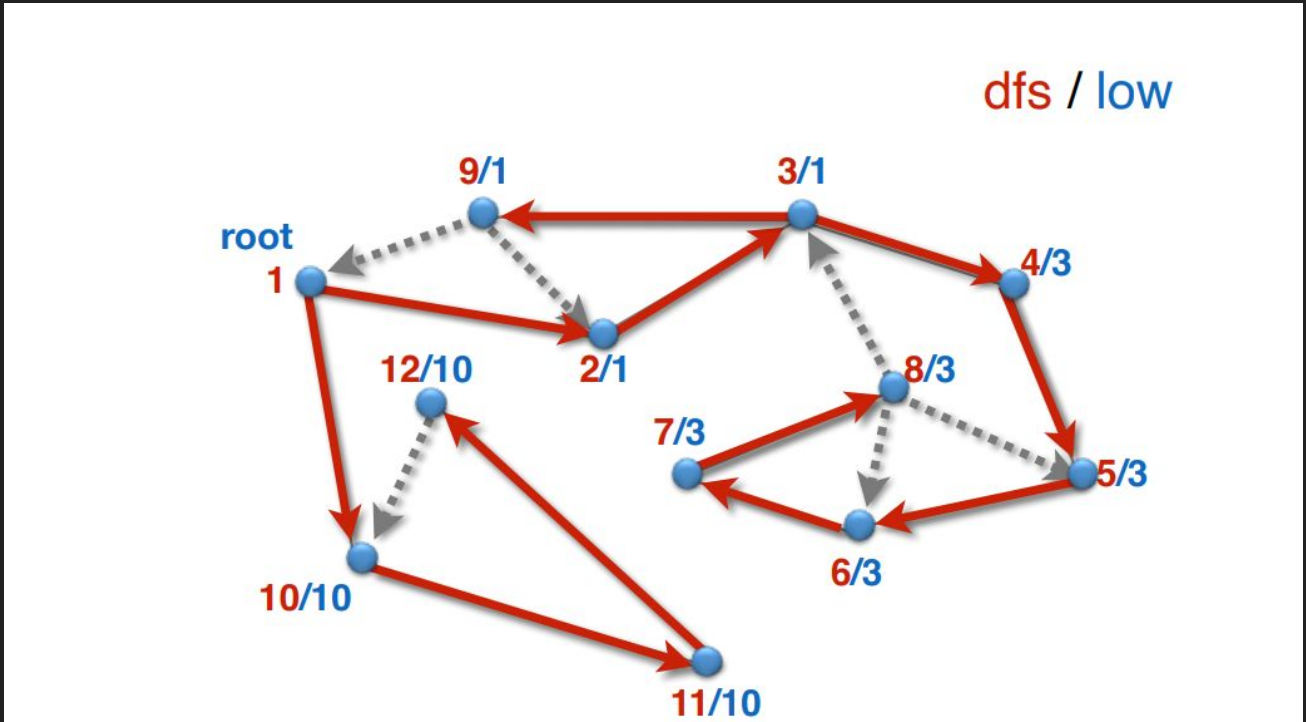
Who are the kids of node with dfs value 4 in dfs tree?



Who are the kids of node with dfs value 8 in dfs tree?



Can the Rest-Edges(Rest-Kanten) be a bridge?



v is an articulation point if and only if

I) v = root and v has degree ≥ 2 in the search tree T

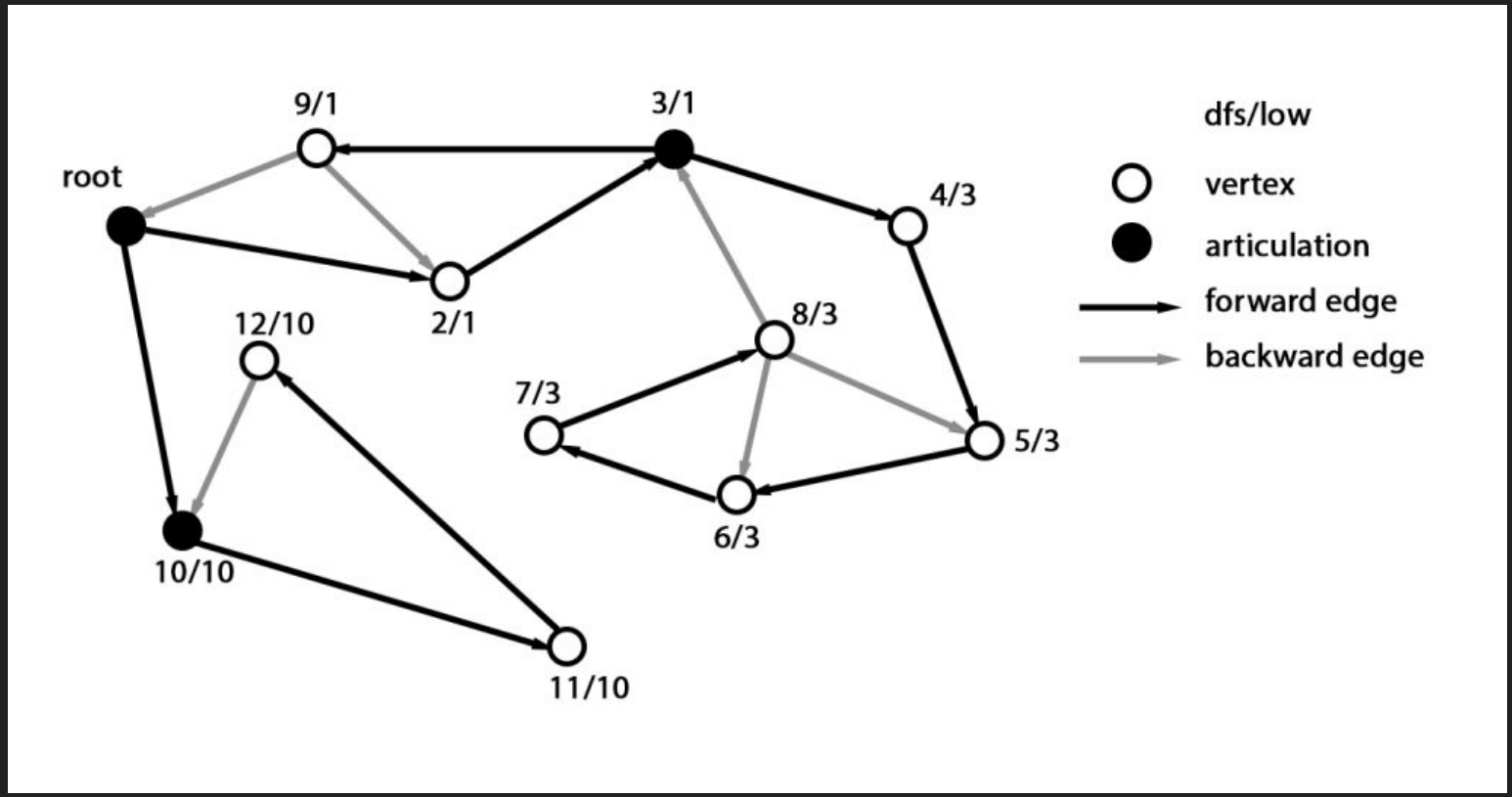
OR

II) v ≠ root and $\exists\, w \in V$ with $\{v, w\} \in E(T)$: low[w] ≥ dfs[v].

$\{u, v\}$ is a bridge if and only if

I) $(u, v) \in E(T) \land \text{low}[v] > \text{Dfs}[u]$

Cut edge = Bridge = Brücke



Eulerian Tour

Closed walk that visits every edge in the graph exactly once

Graph is eulerian(eulersch) if it contains an Eulerian Tour

Eulerian Tour

Closed walk that visits every edge in the graph exactly once

Graph is eulerian(eulersch) if it contains an Eulerian Tour

A connected graph $G = (V, E)$ is Eulerian if and only if all vertices in G have [...] degree.

Eulerian Tour

Closed walk that visits every edge in the graph exactly once

Graph is eulerian(eulersch) if it contains an Eulerian Tour

A connected graph $G = (V, E)$ is Eulerian if and only if all vertices in G have even degree.

Hamiltonian Cycle

finding a cycle in a graph that uses each vertex exactly once.

NP-Complete:

Hamiltonian Cycle

finding a cycle in a graph that uses each vertex exactly once.

NP-Complete: no polynomial time algorithm known, but given a possible solution we can check in polyn. Time whether it’s correct or not

Hamiltonian Cycle

finding a cycle in a graph that uses each vertex exactly once.

NP-Complete: no polynomial time algorithm known, but given a possible solution we can check in polyn. Time whether it's correct or not

A grid graph with m rows and n columns contains a Hamiltonian cycle if and only if $m \cdot n$ is [...].

Hamiltonian Cycle

finding a cycle in a graph that uses each vertex exactly once.

NP-Complete: no polynomial time algorithm known, but given a possible solution we can check in polyn. Time whether it's correct or not

A grid graph with m rows and n columns contains a Hamiltonian cycle if and only if $m \cdot n$ is even.

Hamiltonian Cycle

finding a cycle in a graph that uses each vertex exactly once.

NP-Complete: no polynomial time algorithm known, but given a possible solution we can check in polyn. Time whether it's correct or not

A grid graph with m rows and n columns contains a Hamiltonian cycle if and only if $m \cdot n$ is even.

Hypercubes of any dimension $d \geq 1$ contain a Hamiltonian cycle.