A&D

Exam Preparation

Nil Ozer

Quiz

Outline

- Quiz

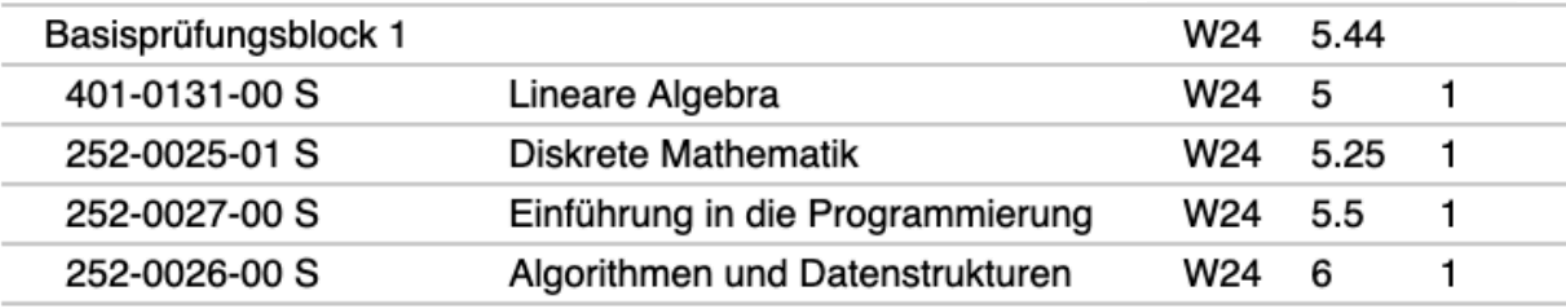
- Lernphase
- Exam Preparation for A&D
    - Website Explanation

- Mock Exam

- Some announcements
- Semester-end Celebration (pizza)

Lernphase

My Basisblock I Grades



Basisprüfungsblock 1		W24	5.44	
401-0131-00 S	Lineare Algebra	W24	5	1
252-0025-01 S	Diskrete Mathematik	W24	5.25	1
252-0027-00 S	Einführung in die Programmierung	W24	5.5	1
252-0026-00 S	Algorithmen und Datenstrukturen	W24	6	1

Lernphase

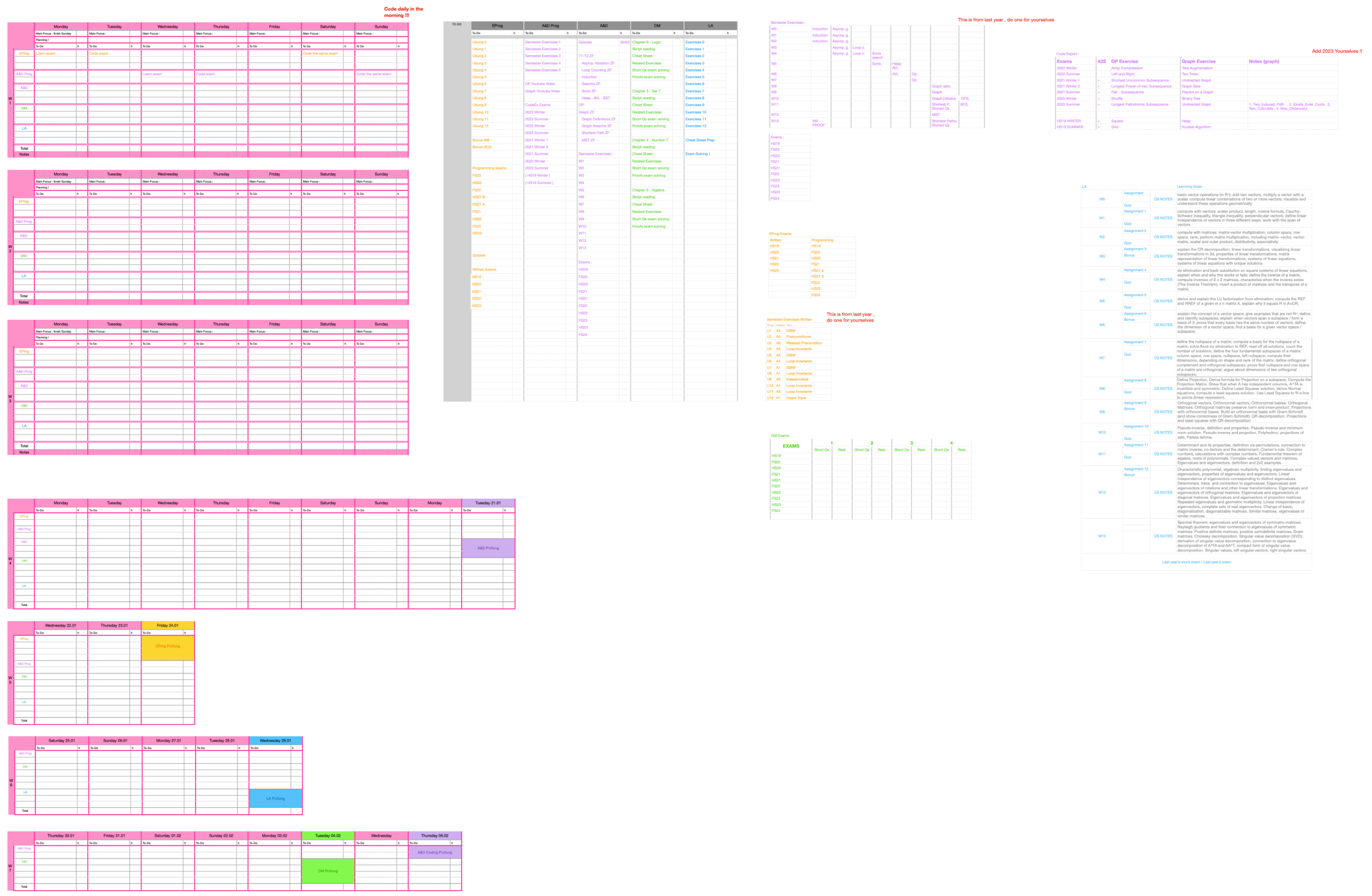
ETH Exams

- Exhausting ! Could be fun :)
- What is “enough” ?
  - How does scale work ?
- Everybody is in the same position as you

Lernphase

Planning

- “Waste” one day on planning !
  - You are making promises to yourself !
- %50 Coding , %50 Theory
  - For now, for the future !
- Interleave the subjects
- My planning looked like …
  - Lernplan template for you !
    - Add/remove necessary

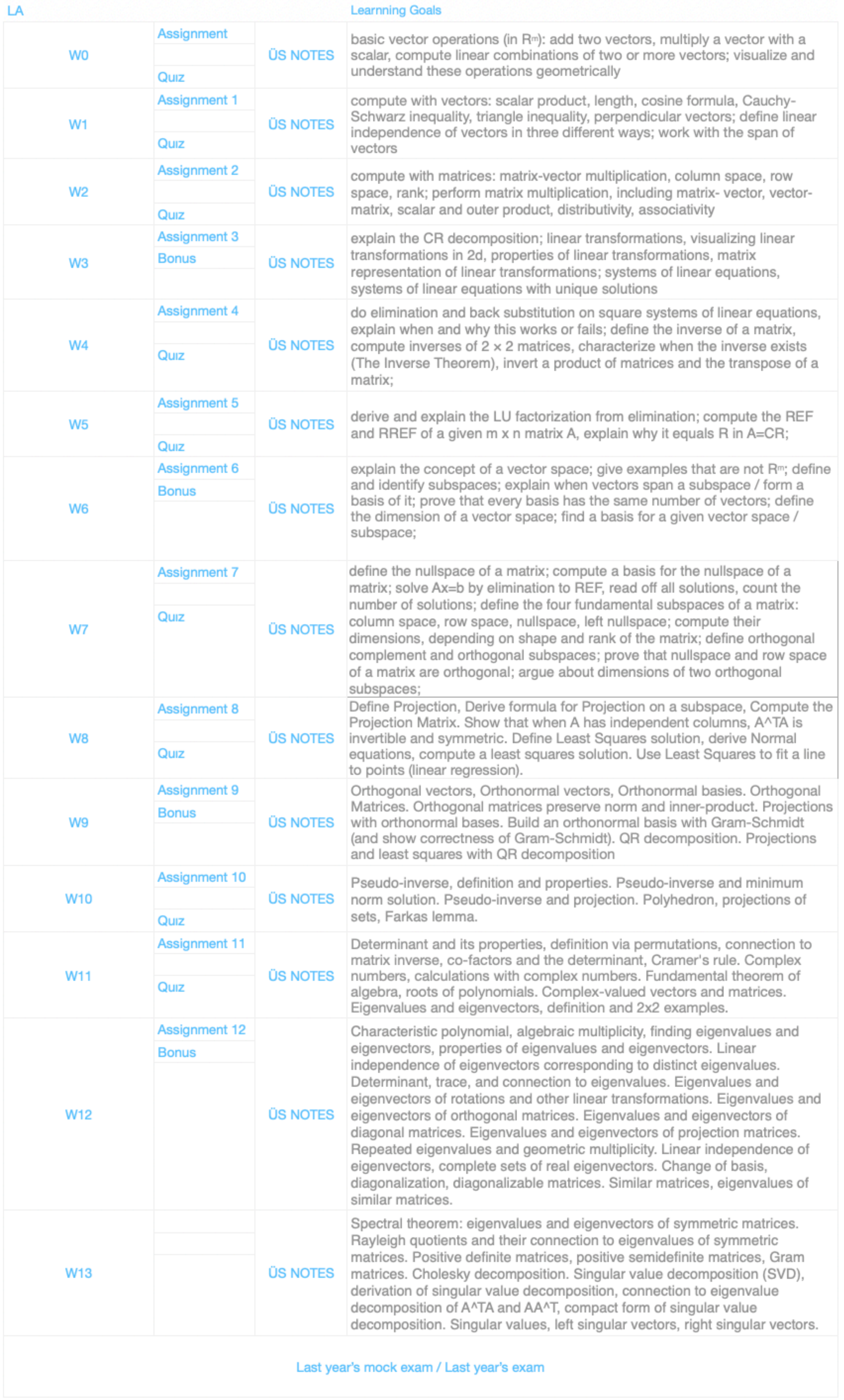


Be aware of your time left at all stages!

Lernphase

Tipps for LA

- Prepare your own CS
  - Don’t waste to much time
  - If so, take over and edit an existing cheat sheet
- Go through the semester exercises
- They take the learning goals, and turn them into questions
- Solve relevant exams

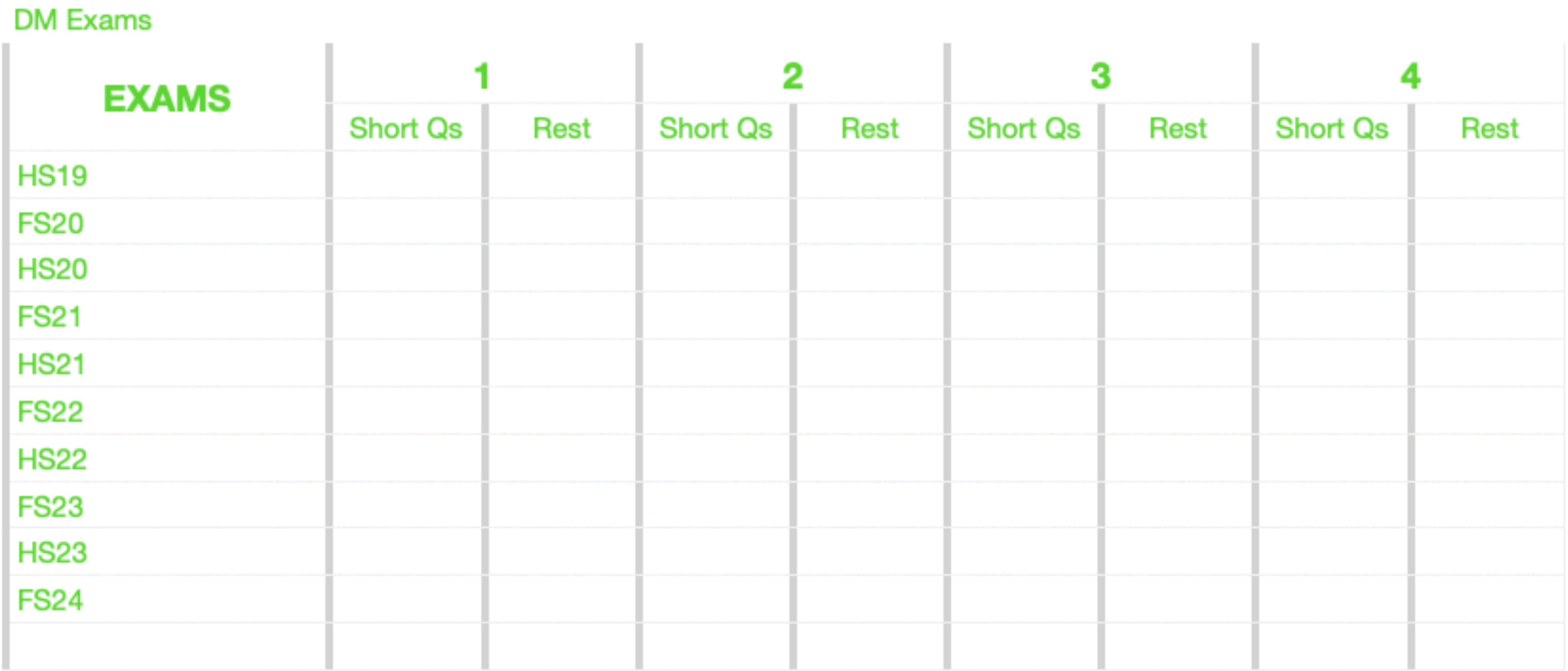


LA			Learning Goals
W0	Assignment Quiz	ÜS NOTES	basic vector operations (in R^m): add two vectors, multiply a vector with a scalar, compute linear combinations of two or more vectors; visualize and understand these operations geometrically
W1	Assignment 1 Quiz	ÜS NOTES	compute with vectors: scalar product, length, cosine formula, Cauchy-Schwarz inequality, triangle inequality, perpendicular vectors; define linear independence of vectors in three different ways; work with the span of vectors
W2	Assignment 2 Quiz	ÜS NOTES	compute with matrices: matrix-vector multiplication, column space, row space, rank; perform matrix multiplication, including matrix- vector, vector-matrix, scalar and outer product, distributivity, associativity
W3	Assignment 3 Bonus	ÜS NOTES	explain the CR decomposition; linear transformations, visualizing linear transformations in 2d, properties of linear transformations, matrix representation of linear transformations; systems of linear equations, systems of linear equations with unique solutions
W4	Assignment 4 Quiz	ÜS NOTES	do elimination and back substitution on square systems of linear equations, explain when and why this works or fails; define the inverse of a matrix, compute inverses of 2 × 2 matrices, characterize when the inverse exists (The Inverse Theorem), invert a product of matrices and the transpose of a matrix;
W5	Assignment 5 Quiz	ÜS NOTES	derive and explain the LU factorization from elimination; compute the REF and RREF of a given m x n matrix A, explain why it equals R in A=CR;
W6	Assignment 6 Bonus	ÜS NOTES	explain the concept of a vector space; give examples that are not R^m; define and identify subspaces; explain when vectors span a subspace / form a basis of it; prove that every basis has the same number of vectors; define the dimension of a vector space; find a basis for a given vector space / subspace;
W7	Assignment 7 Quiz	ÜS NOTES	define the nullspace of a matrix; compute a basis for the nullspace of a matrix; solve Ax=b by elimination to REF, read off all solutions, count the number of solutions; define the four fundamental subspaces of a matrix: column space, row space, nullspace, left nullspace; compute their dimensions, depending on shape and rank of the matrix; define orthogonal complement and orthogonal subspaces; prove that nullspace and row space of a matrix are orthogonal; argue about dimensions of two orthogonal subspaces;
W8	Assignment 8 Quiz	ÜS NOTES	Define Projection, Derive formula for Projection on a subspace, Compute the Projection Matrix. Show that when A has independent columns, A^TA is invertible and symmetric. Define Least Squares solution, derive Normal equations, compute a least squares solution. Use Least Squares to fit a line to points (linear regression).
W9	Assignment 9 Bonus	ÜS NOTES	Orthogonal vectors, Orthonormal vectors, Orthonormal basies. Orthogonal Matrices. Orthogonal matrices preserve norm and inner-product. Projections with orthonormal bases. Build an orthonormal basis with Gram-Schmidt (and show correctness of Gram-Schmidt). QR decomposition. Projections and least squares with QR decomposition
W10	Assignment 10 Quiz	ÜS NOTES	Pseudo-inverse, definition and properties. Pseudo-inverse and minimum norm solution. Pseudo-inverse and projection. Polyhedron, projections of sets, Farkas lemma.
W11	Assignment 11 Quiz	ÜS NOTES	Determinant and its properties, definition via permutations, connection to matrix inverse, co-factors and the determinant, Cramer's rule. Complex numbers, calculations with complex numbers. Fundamental theorem of algebra, roots of polynomials. Complex-valued vectors and matrices. Eigenvalues and eigenvectors, definition and 2x2 examples.
W12	Assignment 12 Bonus	ÜS NOTES	Characteristic polynomial, algebraic multiplicity, finding eigenvalues and eigenvectors, properties of eigenvalues and eigenvectors. Linear independence of eigenvectors corresponding to distinct eigenvalues. Determinant, trace, and connection to eigenvalues. Eigenvalues and eigenvectors of rotations and other linear transformations. Eigenvalues and eigenvectors of orthogonal matrices. Eigenvalues and eigenvectors of diagonal matrices. Eigenvalues and eigenvectors of projection matrices. Repeated eigenvalues and geometric multiplicity. Linear independence of eigenvectors, complete sets of real eigenvectors. Change of basis, diagonalization, diagonalizable matrices. Similar matrices, eigenvalues of similar matrices.
W13		ÜS NOTES	Spectral theorem: eigenvalues and eigenvectors of symmetric matrices. Rayleigh quotients and their connection to eigenvalues of symmetric matrices. Positive definite matrices, positive semidefinite matrices, Gram matrices. Cholesky decomposition. Singular value decomposition (SVD), derivation of singular value decomposition, connection to eigenvalue decomposition of A^TA and AA^T, compact form of singular value decomposition. Singular values, left singular vectors, right singular vectors.
Last year’s mock exam / Last year’s exam			

Lernphase

Tipps for DM

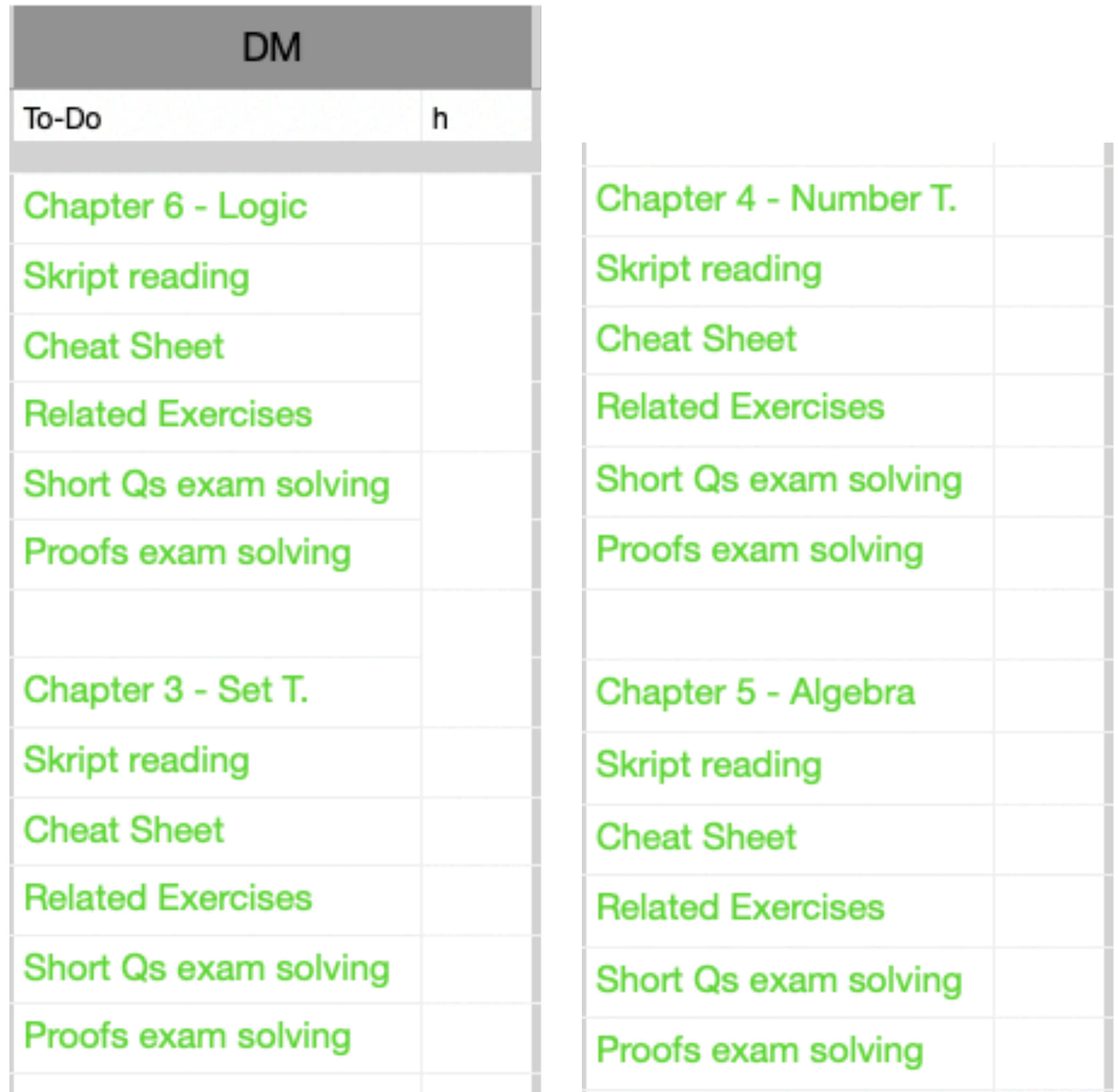
- Take over and edit an existing cheat sheet
- Learn chapter by chapter
  - My advice: Logic, Set, Number, Algebra
  - Also solve like this in the exam !!



DM Exams

EXAMS	1		2		3		4	
	Short Qs	Rest	Short Qs	Rest	Short Qs	Rest	Short Qs	Rest
HS19								
FS20								
HS20								
FS21								
HS21								
FS22								
HS22								
FS23								
HS23								
FS24								

- For each chapter do these ->



DM

To-Do	h
Chapter 6 - Logic	
Skript reading	
Cheat Sheet	
Related Exercises	
Short Qs exam solving	
Proofs exam solving	
Chapter 3 - Set T.	
Skript reading	
Cheat Sheet	
Related Exercises	
Short Qs exam solving	
Proofs exam solving	

Chapter 4 - Number T.	
Skript reading	
Cheat Sheet	
Related Exercises	
Short Qs exam solving	
Proofs exam solving	
Chapter 5 - Algebra	
Skript reading	
Cheat Sheet	
Related Exercises	
Short Qs exam solving	
Proofs exam solving	

- Then resolve the exams
  - Have complete exam trials

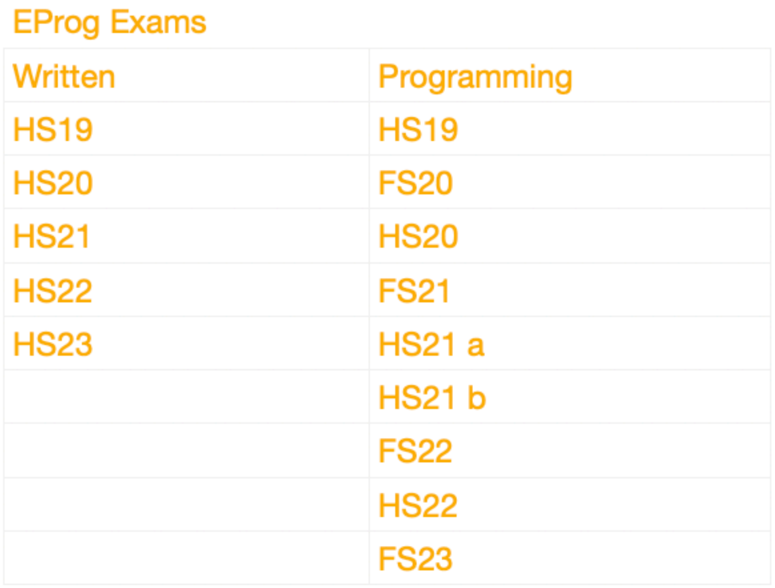
Lernphase

Tipps for EProg

- If you don't have the idea immediately yet, don't worry !
  - Timed bonus, don't worry !!

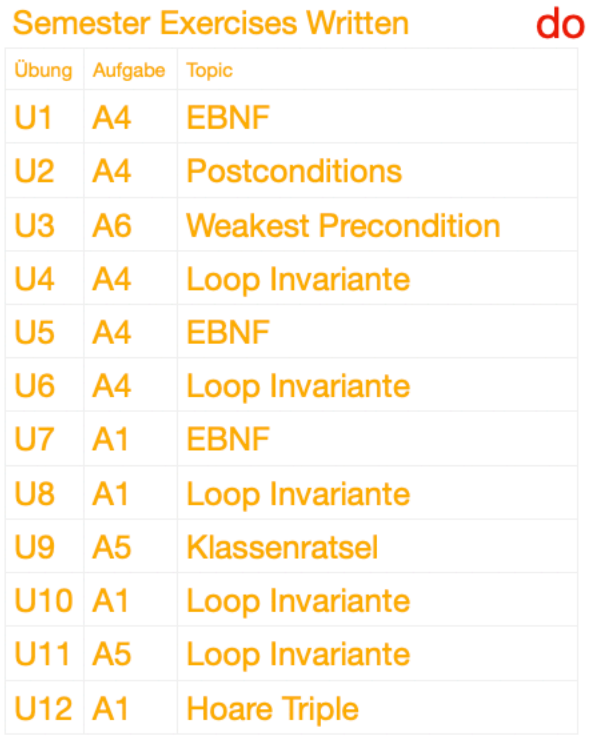
- Start by learning the question
  - Read the question, look at the solutions, understand it !
- Then code it yourself without looking
- Then try it by yourself on another day !

- Don't underestimate the written exams !! Tricky



EProg Exams

Written	Programming
HS19	HS19
HS20	FS20
HS21	HS20
HS22	FS21
HS23	HS21 a
	HS21 b
	FS22
	HS22
	FS23



Semester Exercises Written

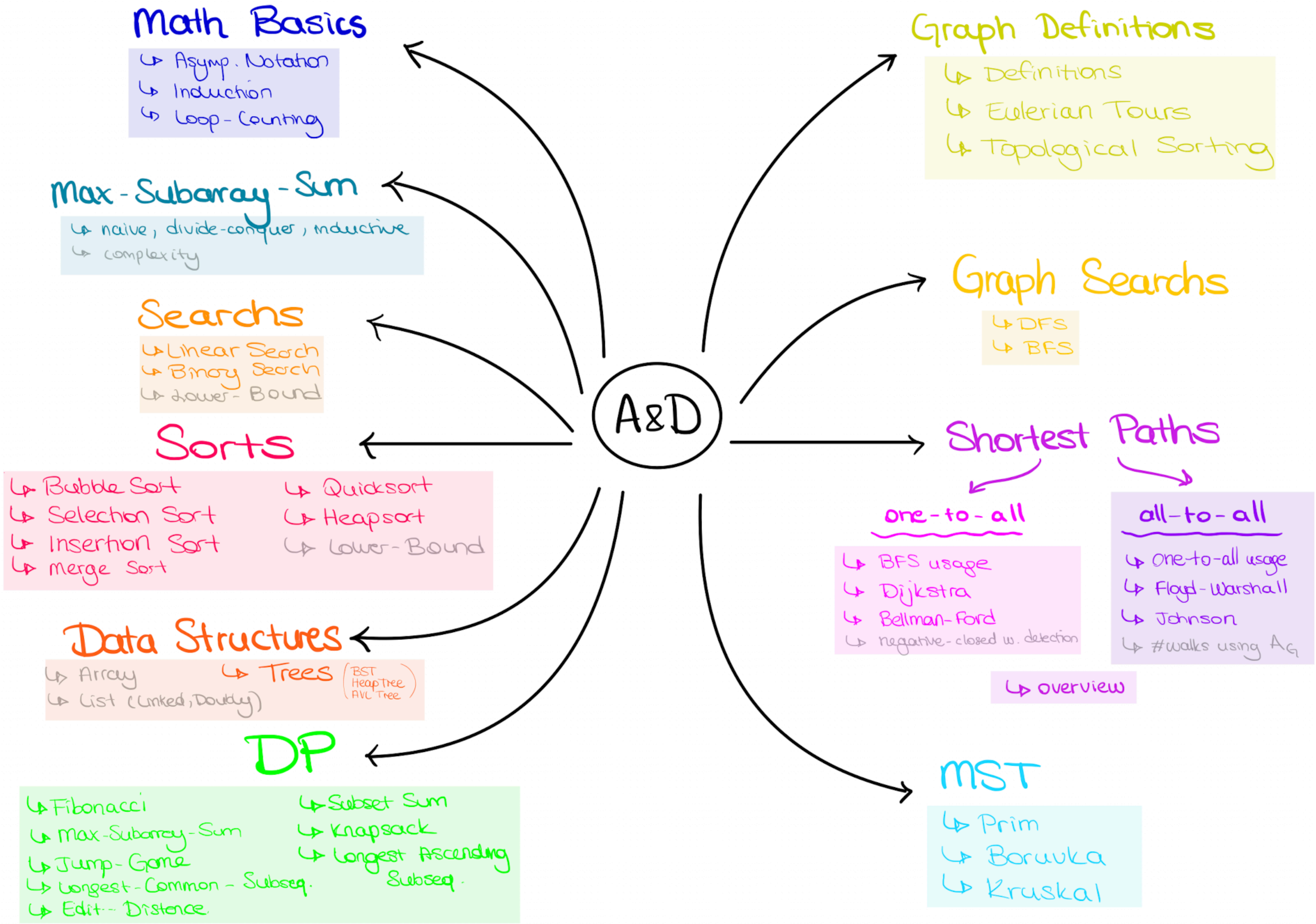
Übung	Aufgabe	Topic
U1	A4	EBNF
U2	A4	Postconditions
U3	A6	Weakest Precondition
U4	A4	Loop Invariante
U5	A4	EBNF
U6	A4	Loop Invariante
U7	A1	EBNF
U8	A1	Loop Invariante
U9	A5	Klassenratsel
U10	A1	Loop Invariante
U11	A5	Loop Invariante
U12	A1	Hoare Triple

This is from last year ,
do one for yourselves

A&D Exam

I’ve got you !

A&D Overview

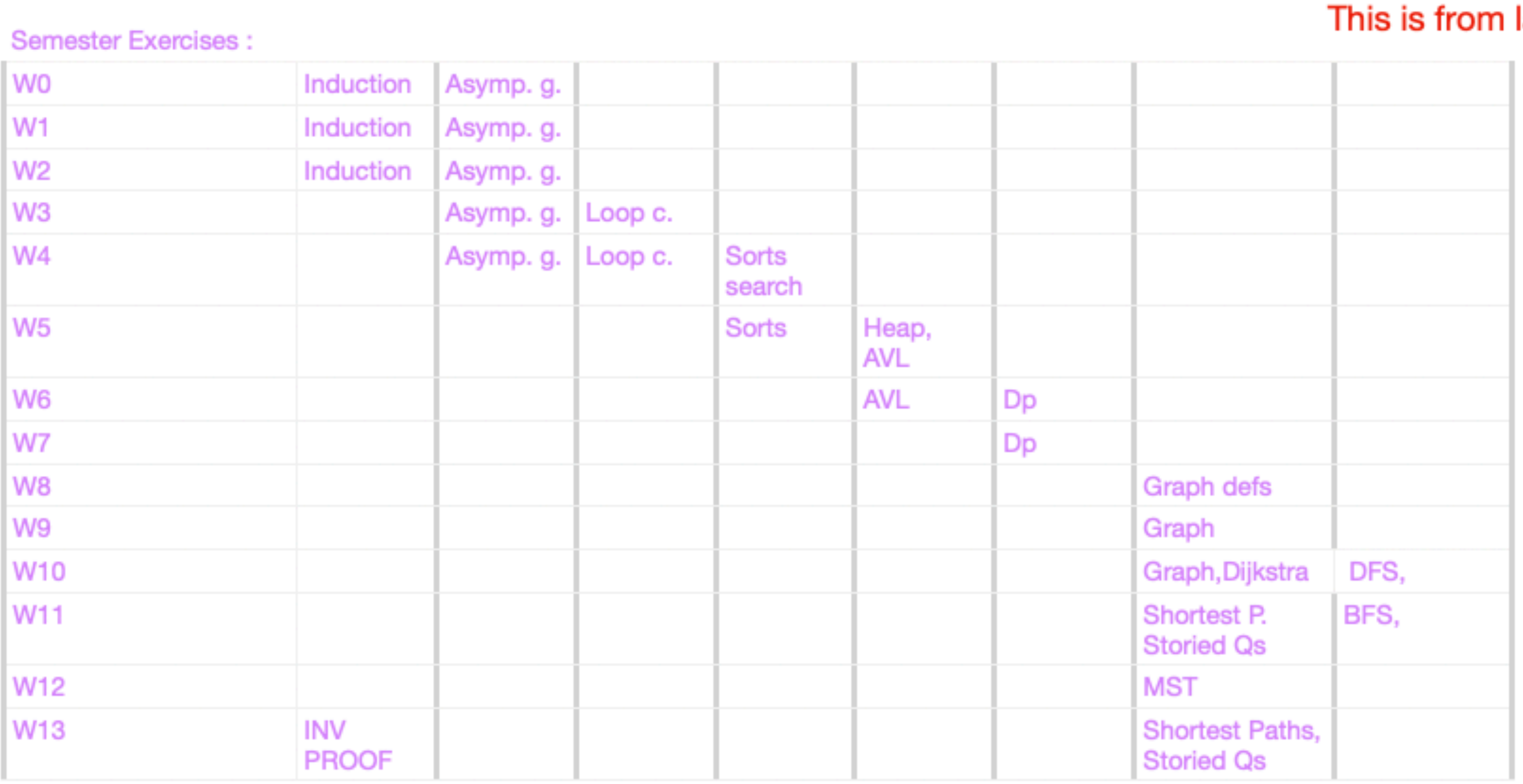

A&D
Math Basics
↳ Asymp. Notation
↳ Induction
↳ Loop-Counting
Max-Subarray-Sum
↳ naive, divide-conquer, inductive
↳ complexity
Searchs
↳ Linear Search
↳ Binary Search
↳ Lower-Bound
Sorts
↳ Bubble Sort
↳ Selection Sort
↳ Insertion Sort
↳ merge Sort
↳ Quicksort
↳ Heapsort
↳ Lower-Bound
Data Structures
↳ Array
↳ List (Linked, Doubly)
↳ Trees
BST
Heap Tree
AVL Tree
DP
↳ Fibonacci
↳ Max-Subarray-Sum
↳ Jump-Game
↳ Longest-Common-Subseq.
↳ Edit-Distance
↳ Subset Sum
↳ Knapsack
↳ Longest Ascending Subseq.
Graph Definitions
↳ Definitions
↳ Eulerian Tours
↳ Topological Sorting
Graph Searchs
↳ DFS
↳ BFS
Shortest Paths
one-to-all
↳ BFS usage
↳ Dijkstra
↳ Bellman-Ford
↳ negative-closed w. detection
all-to-all
↳ One-to-all usage
↳ Floyd-Warshall
↳ Johnson
↳ #walks using A_G
↳ overview
MST
↳ Prim
↳ Boruvka
↳ Kruskal

Lernphase

Tipps for A&D

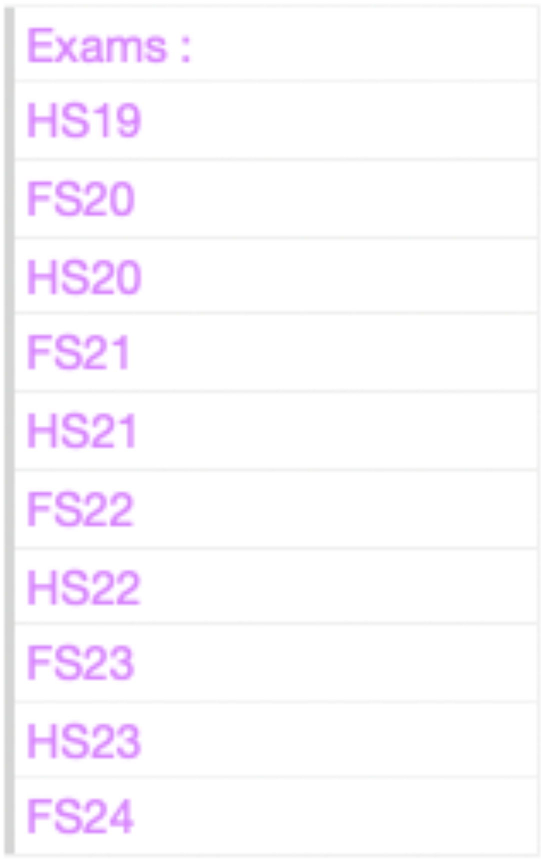
- As I promised, you’re very well prepared already
- However, you need to practice ! There’s a lot to do !

This is from last year , do one for yourselves



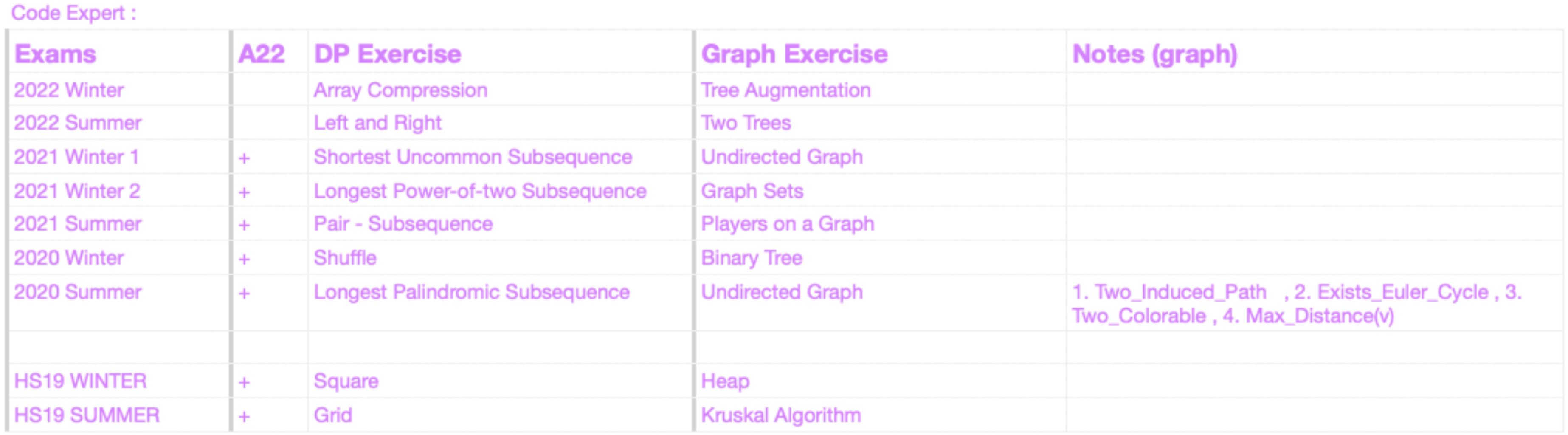
Semester Exercises :

W0	Induction	Asymp. g.						
W1	Induction	Asymp. g.						
W2	Induction	Asymp. g.						
W3		Asymp. g.	Loop c.					
W4		Asymp. g.	Loop c.	Sorts search				
W5				Sorts	Heap, AVL			
W6					AVL	Dp		
W7						Dp		
W8							Graph defs	
W9							Graph	
W10							Graph,Dijkstra	DFS,
W11							Shortest P. Storied Qs	BFS,
W12							MST	
W13	INV PROOF						Shortest Paths, Storied Qs	



Exams :
HS19
FS20
HS20
FS21
HS21
FS22
HS22
FS23
HS23
FS24

Add 2023 Yourselves !!



Code Expert :

Exams	A22	DP Exercise	Graph Exercise	Notes (graph)
2022 Winter		Array Compression	Tree Augmentation	
2022 Summer		Left and Right	Two Trees	
2021 Winter 1	+	Shortest Uncommon Subsequence	Undirected Graph	
2021 Winter 2	+	Longest Power-of-two Subsequence	Graph Sets	
2021 Summer	+	Pair - Subsequence	Players on a Graph	
2020 Winter	+	Shuffle	Binary Tree	
2020 Summer	+	Longest Palindromic Subsequence	Undirected Graph	1. Two_Induced_Path , 2. Exists_Euler_Cycle , 3. Two_Colorable , 4. Max_Distance(v)
HS19 WINTER	+	Square	Heap	
HS19 SUMMER	+	Grid	Kruskal Algorithm	

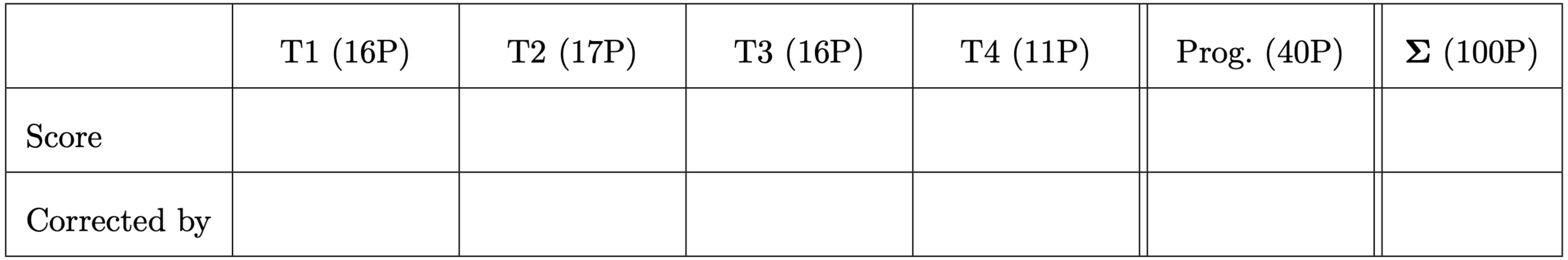
Exam

Written Exam

- %60
- 2h
- VIS , Mock Exam

Programming Exam

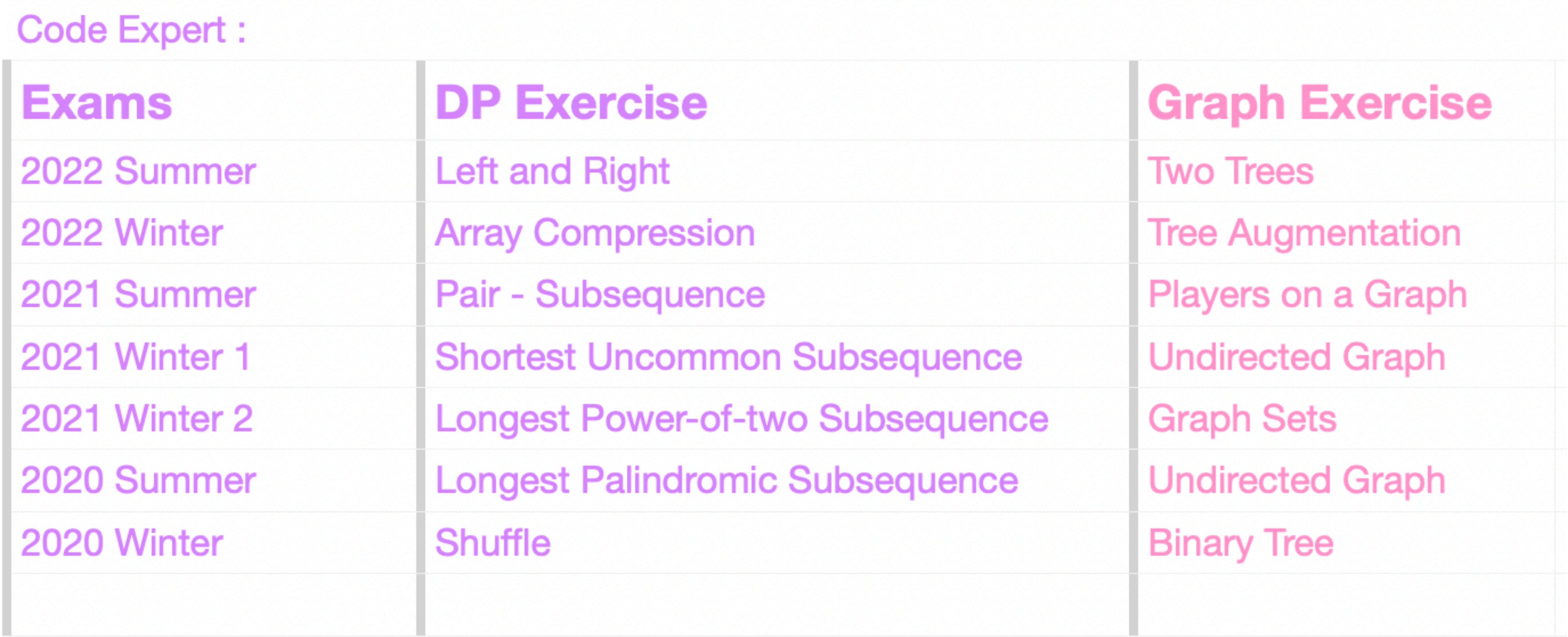
- %40
- 3h
- CodeExpert



	T1 (16P)	T2 (17P)	T3 (16P)	T4 (11P)	Prog. (40P)	Σ (100P)
Score						
Corrected by						

Programming Exam

- One DP one Graph Exercise each year
- Old exams will be published at the end of the semester

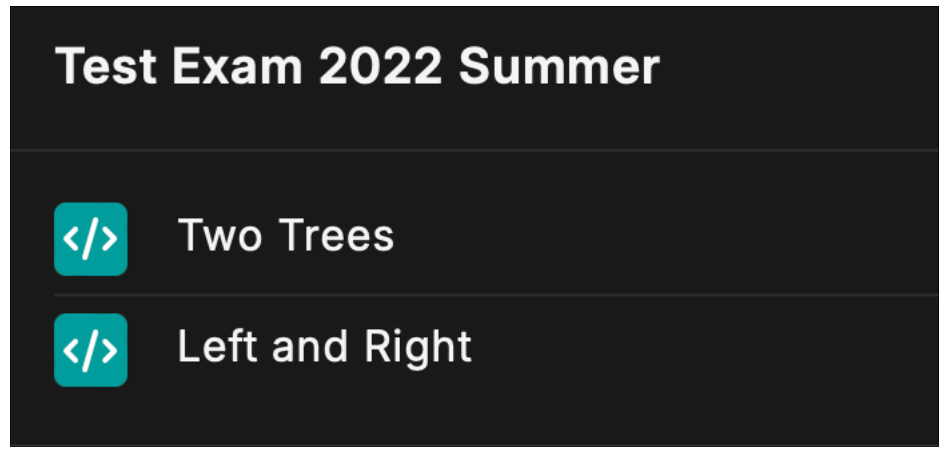


Code Expert :

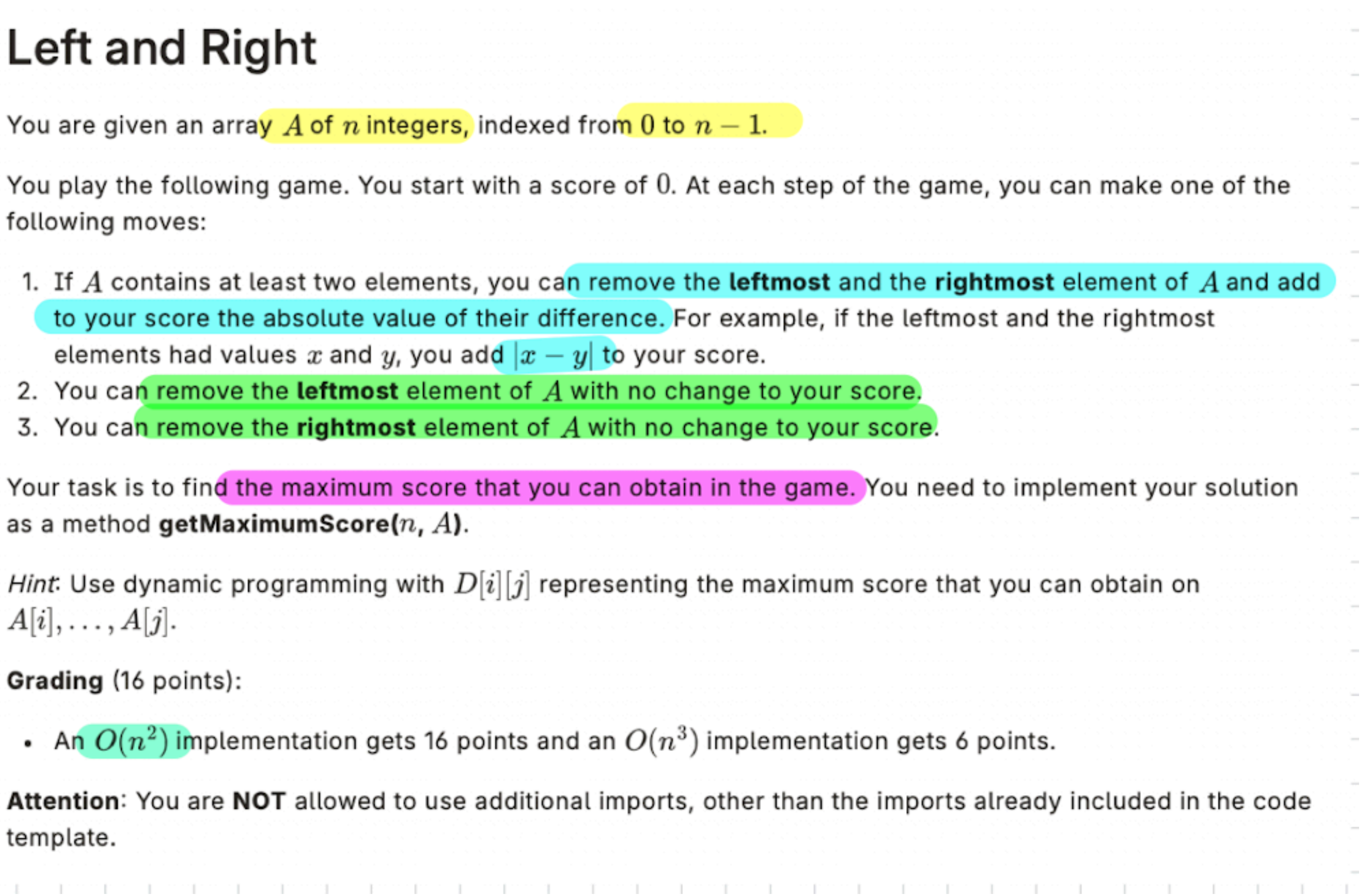
Exams	DP Exercise	Graph Exercise
2022 Summer	Left and Right	Two Trees
2022 Winter	Array Compression	Tree Augmentation
2021 Summer	Pair - Subsequence	Players on a Graph
2021 Winter 1	Shortest Uncommon Subsequence	Undirected Graph
2021 Winter 2	Longest Power-of-two Subsequence	Graph Sets
2020 Summer	Longest Palindromic Subsequence	Undirected Graph
2020 Winter	Shuffle	Binary Tree

If there are any changes, you will see !

Programming Exam



Test Exam 2022 Summer



Left and Right

You are given an array A of n integers, indexed from 0 to $n-1$.

You play the following game. You start with a score of 0. At each step of the game, you can make one of the following moves:

1. If A contains at least two elements, you can remove the **leftmost** and the **rightmost** element of A and add to your score the absolute value of their difference. For example, if the leftmost and the rightmost elements had values x and y, you add $|x-y|$ to your score.
2. You can remove the **leftmost** element of A with no change to your score.
3. You can remove the **rightmost** element of A with no change to your score.

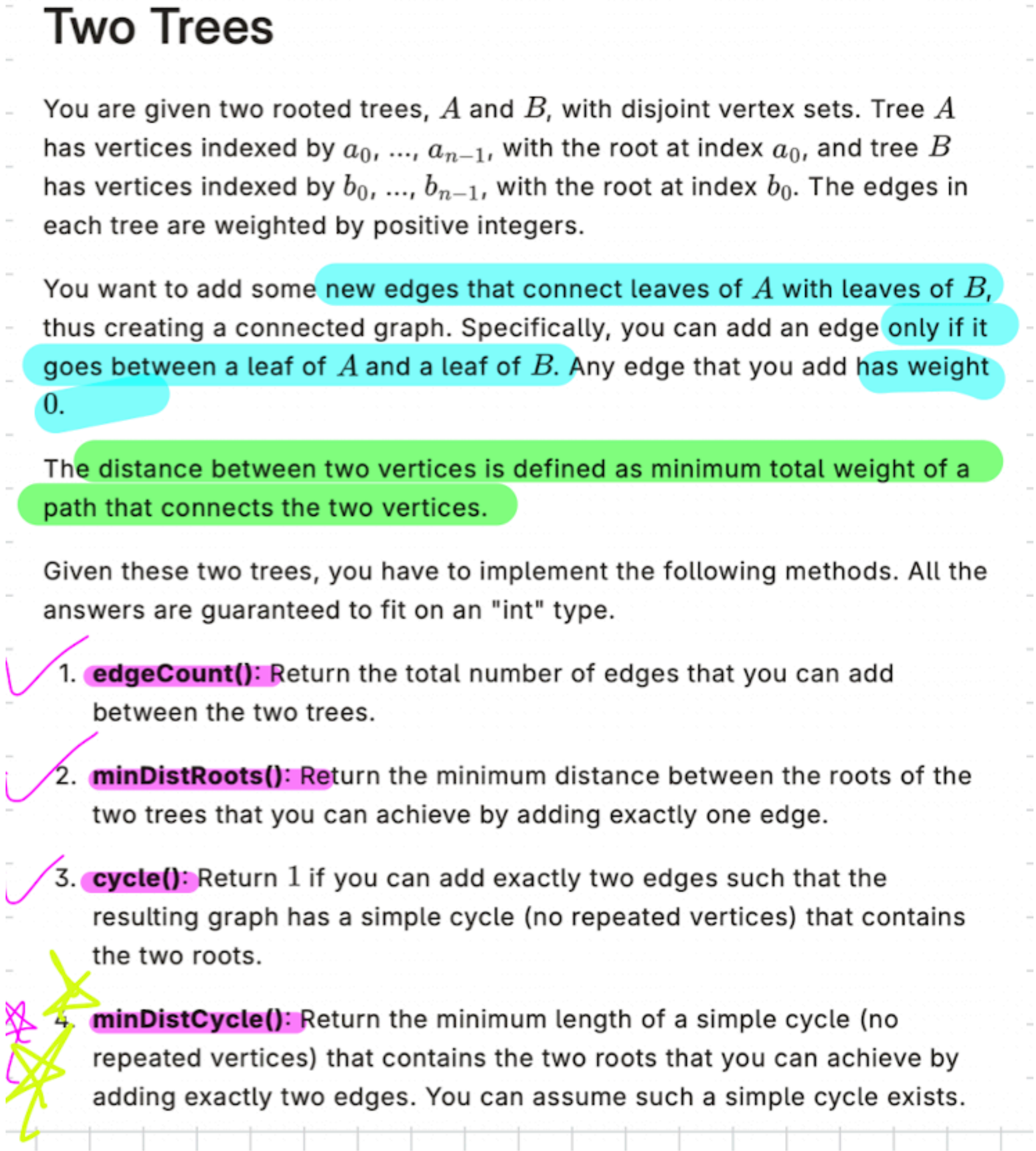
Your task is to find the maximum score that you can obtain in the game. You need to implement your solution as a method **getMaximumScore(n, A)**.

Hint: Use dynamic programming with $D[i][j]$ representing the maximum score that you can obtain on $A[i], \ldots, A[j]$.

Grading (16 points):

- An $O(n^2)$ implementation gets 16 points and an $O(n^3)$ implementation gets 6 points.

Attention: You are **NOT** allowed to use additional imports, other than the imports already included in the code template.



Two Trees

You are given two rooted trees, A and B, with disjoint vertex sets. Tree A has vertices indexed by a_0, ..., a_{n-1}, with the root at index a_0, and tree B has vertices indexed by b_0, ..., b_{n-1}, with the root at index b_0. The edges in each tree are weighted by positive integers.

You want to add some new edges that connect leaves of A with leaves of B, thus creating a connected graph. Specifically, you can add an edge only if it goes between a leaf of A and a leaf of B. Any edge that you add has weight 0.

The distance between two vertices is defined as minimum total weight of a path that connects the two vertices.

Given these two trees, you have to implement the following methods. All the answers are guaranteed to fit on an "int" type.

1. **edgeCount()**: Return the total number of edges that you can add between the two trees.
2. **minDistRoots()**: Return the minimum distance between the roots of the two trees that you can achieve by adding exactly one edge.
3. **cycle()**: Return 1 if you can add exactly two edges such that the resulting graph has a simple cycle (no repeated vertices) that contains the two roots.
4. **minDistCycle()**: Return the minimum length of a simple cycle (no repeated vertices) that contains the two roots that you can achieve by adding exactly two edges. You can assume such a simple cycle exists.

Written Exam

Old Structure

- T1 : Basics I
- T2 : Basics II
- T3 : DP
- T4 : Graph

New Structure

- T1 : Complexity and O-Notation
- T2 : Graphs
- T3 : Algorithm Design
- T4 : Proofs

Tasks are always the same, just grouped differently !

Graph Modelling ✗ with a,b,c,d,e

One Graph Modelling task in T3

Written Exam

- T1 : Complexity and O-Notation
  - Asymp. Notation Quiz
  - Loop Counting
  - Sorting/Searching Algos Quiz
- T2 : Graphs
  - Graph Quiz
  - Binary Trees (BST, AVL, Heap)
  - BFS/DFS
  - MST
  - SPT
- T3 : Algortihm Design
  - DP
  - Graph Modelling
  - “Formal” graph algo
  - Additional one
- T4 : Proofs
  - Induction
  - Graph proofs

Exam Preparation Page Introduction

How to study for A&D

in the Lernphase

Use the exam prep page !!

- Recap topics use the summary/skript , Add/remove things , prepare your own summary !
- Exercise Sheets solve the relevant exercises not the weird ones
- Exam task examples document with all exam tasks we’ve solved in class , try it yourself !
- Exams Solve most of them ! They are still highly relevant !
- You can always ask me :)
- Watch the youtube videos when I upload them
- Work on DP tasks, Work on Graph tasks (BFS,DFS,Dijkstra uploaded)
- Do exam trials

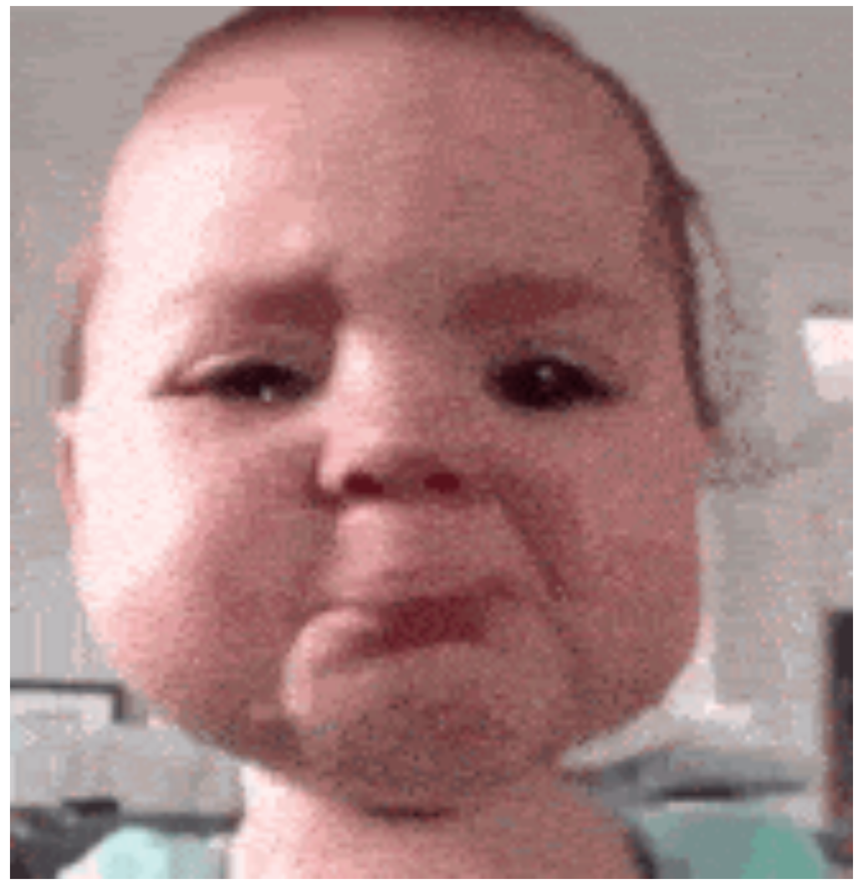
A&D 😊🧑‍💻

WhatsApp group

Mock Exam

Some announcements

Bye...



Nil Ozer