Parameterized Algorithms
Summer Term 2026
`scale.iti.kit.edu`

Karlsruher Institut für Technologie

Exercise Sheet 6

Submission due by July 16, 2026

Problem 1: Dominating Set on Special Graphs — 10 points

Consider the following parameterized variant of Dominating Set. The graph G is given together with an ordering of the vertices $V = \{v_1, \ldots, v_n\}$. The *length* of an edge $\{v_i, v_j\}$ is $|i-j|$. The parameter k is the maximum edge length in G. Provide an FPT algorithm that solves this parameterization of Dominating Set.

Problem 2: Miscellaneous facts about treewidth — 6 + 6 points

Part (a) Let X_i, X_k, and X_j be bags of a tree decomposition of a graph G such that X_k lies on the path between X_i and X_j. Show that $X_i \setminus X_k$ is separated from $X_j \setminus X_k$ by X_k.

Part (b) Let G be a graph of treewidth at most k with n vertices and m edges. Show that $m \leq n \cdot k$.

Problem 3: Connected Dominating Sets — 18 points

A vertex set $D \subseteq V$ in a graph $G = (V,E)$ is a *dominating set* if every vertex $v \in V$ is either contained in D or has a neighbor in D. The Connected Dominating Set problem asks for a smallest possible dominating set D such that the subgraph induced by D is connected.

Given a graph G, a parameter k, and a tree decomposition of width k, provide an FPT algorithm for this parameterization of Connected Dominating Set.

Hint: Use dynamic programming on the tree decomposition. Using Courcelle's Theorem (from lecture 10) is not allowed.

Problem 4: Treewidth DP — 10 bonus points

In this exercise, you have to implement a dynamic program on a tree decomposition to solve the Independent Set problem. On the course website, you will find five instances, each consisting of a graph and a corresponding tree decomposition.

State the size of a maximum independent set for each instance and briefly explain how you realized the lecture's approach and any optimizations you applied.

Input format of a graph (`.gr`): The first line `p tw n m` contains the letter `p`, the word `tw`, the number of vertices `n`, and the number of edges `m`. The following `m` lines each describe an edge;

each line contains two numbers `a b` with $1 \leq a,b \leq n$, denoting an edge between vertex `a` and vertex `b`. Since the graphs are undirected, each edge is defined only once.

Example:

```
p tw 5 6
1 2
1 3
2 3
2 4
3 4
2 5
```

Input format of a tree decomposition (`.td`): The first line starts with `s td N W n`, where `N` is the number of bags, `W` the maximum bag size, and `n` the original graph's vertex count. The following `N` lines have the form `b i v1 v2 ...`, where `i` is the bag number ($1 \leq i \leq N$) and `v1`, ... are the vertices contained in that bag. All remaining lines define the edges of the tree decomposition; each line contains two integers `i j` ($1 \leq i < j \leq N$), denoting an edge between bag `i` and bag `j`.

Example:

```
s td 3 3 5
b 1 1 2 3
b 2 2 3 4
b 3 2 5
1 2
2 3
```