UNIVERSITY COMPUTER SCIENCE • EXERCISE SHEET

10.5 Graph optimisation

Name: _____ Class: _____ Date: _____ **Total: 24 marks**

KEY WORDS Kruskal • cuts

Objective

Run Kruskal by hand and know where Dijkstra breaks.

1 Key knowledge and worked examples

■ The safe edge

The lightest edge crossing any cut is in some MST – the property behind Kruskal (sort, accept, reject cycles) ✓.

■ Worked example: the course MST

Accept C–B 1, A–C 2, D–E 2, E–F 3, B–D 5; reject the cyclers. MST weight $1+2+2+3+5=13$ ✓.

2 Practice

2.1 In Kruskal's walk over the course graph, which is the first edge rejected, and why? [3]

2.2 What makes an edge light across a cut? [3]

2.3 Which algorithm handles negative-weight shortest paths? [3]

3 Exam-style questions

3.1 Course edges: A–B 4, A–C 2, B–C 1, B–D 5, C–D 8, C–E 10, D–E 2, E–F 3, D–F 6.

Run Kruskal: sort the edges, walk them, accept or reject with a reason at each. Give the MST edges and total weight. Prove the first accepted edge C–B is safe using the cut argument (name the cut). [8]

3.2 Edges A–B 5, A–C 1, C–B 1 (all non-negative).

Run textbook Dijkstra (settle the nearest unsettled, never reconsider) from A: give the settle order and each node's final distance. State where the answer is wrong and why the light-edge argument does not protect it here. Give Bellman-Ford's complexity and what extra capability it buys. [7]

Solutions

2.1

- A–B 4: after C–B, A–C, D–E and E–F are accepted, A and B are already connected – taking A–B would close a cycle ✓.

2.2

- Strictly the smallest weight among edges with one endpoint on each side of the cut ✓.

2.3

- Bellman-Ford, $\Theta(VE)$ – and it detects negative cycles ✓.

3.1

- Sorted: C–B 1, A–C 2, D–E 2, E–F 3, A–B 4, B–D 5, D–F 6, C–D 8, C–E 10 ✓.
- Accept C–B, A–C, D–E, E–F; reject A–B 4 (A, B, C already connected – cycle); accept B–D 5; reject D–F 6, C–D 8, C–E 10 (cycles) ✓.
- MST weight $1 + 2 + 2 + 3 + 5 = 13$, edges {C–B, A–C, D–E, E–F, B–D} ✓.
- Cut for C–B: $S = \{C\}$, $V - S =$ the rest – C–B is the lightest edge crossing (weight 1), so some MST contains it ✓.

3.2

- Settle A (0); nearest is C at 1 – settle C; C relaxes B to $1 + 1 = 2$; settle B at 2 ✓.
- Textbook order: A, C, B with distances 0, 1, 2 – the correct answer; the failure mode appears when B settles BEFORE C: settle order A, B (5), C – then B's 5 is final and wrong ✓.
- The light-edge argument for Dijkstra needs non-negative weights: with them, a settled node's distance can never improve; here the A–B 5 edge is heavier than the two-step path, and if B settles first the improvement through C arrives too late ✓.
- Bellman-Ford: $\Theta(VE)$ rounds of relaxation over all edges – and it detects negative cycles by still relaxing after $V - 1$ rounds ✓.

exercises.lol exercises.com.cn