Software Testing 2025/2026 – Test Performance (LO4)

<u>R1.1</u>

Adequacy Criteria	Target	Actual
Statement coverage	>95%	100%
Branch coverage	>95%	100%

Target justification - Validation logic is rule-heavy and branch-dense, meaning near-complete exhaustion of logic is feasible and is a valid judge of adequacy.

Improvements to meet targets – N/A

<u>R1.2</u>

Adequacy Criteria	Target	Actual
Constraint satisfaction	100% of cases	100% of cases
Map topology classes covered	6-8 classes	7 classes
Random map tests	>50 runs	0 runs

Target justification – There is no explicit finite coverage metric for path generation in potentially infinite maps, so adequacy of testing can be defined by high coverage of behaviours and constraints, as well as success in repeated random tests (for probabilistic fault detection).

Improvements to meet targets – Instrumentation to generate random maps would allow probabilistic fault detection to improve confidence in testing

<u>R2.1</u>

Adequacy Criteria	Target	Actual
Worst-case runtime	<60s	354.281s
95th percentile	<45s	42.19742s
Median runtime	<10s	0.288s
Time complexity with NFZs	≤O(n)	O(n) (see tables in Test Evidence)

Target justification – While these criteria directly describe performance, they are used to show that the performance testing covers a wide range of scenarios (coverage of the *performance* space) such as typical user experience using the median, and time complexity to ensure there's no exponential blowup. It covers this to the point the testing can confirm R2.1 has been met in realistic cases.

Improvements to meet targets – The worst-case runtime target does not need to be met, as it exists as an expected metric for the system. This target not being met provides performance information to aid in system analysis and is a flaw on the part of the system, not the testing.