

Phase 2 Grading Sheet, CSE535, Scott Stoller, Stony Brook U, 2017-09-12		POSSIBLE	ACTUAL
Team:		POINTS	POINTS
CLIENT			
generate pseudorandom workload with good diversity using specified seed		4	
generate request sequence specified in config file		4	
handle result: check signatures and hashes in result proof		6	
timeout and send request to all replicas if timely response not received		6	
check that dictionary contains expected content at end of test case		4	
OLYMPUS			
create initial configuration: create keys, create, setup, and start processes		10	
REPLICA			
dictionary object: support put, get, slice, append		6	
head: handle request: assign slot, sign order stmt and result stmt, send shuttle		6	
handle shuttle: check validity of order proof (incl. signatures), add signed order statement and signed result statement, send updated shuttle		12	
handle result shuttle: validate, save, and forward it		8	
non-head: handle request: send cached result, send error, or forward request		6	
fault-injection: required triggers (1 pt each)		4	
fault-injection: required failures (1 pt each)		3	
MULTI-HOST EXECUTION			
processes are spread across multiple hosts		3	
CONFIGURATION FILES			
support configuration files specified in project.txt (if not, causing inability to test some functionality with instructor's testcases, also lose 1/3 of points for it)		4	
LOGS			
detailed and readable logs (If some functionality cannot be adequately verified due to inadequate logs, deduct suitable points for each affected item above)		6	
DOCUMENTATION			
README and testing.txt contain all information specified in project.txt		8	
PENALTIES			
bugs/limitations not reported in README (additional 50% off)			
poor code quality (up to -10, depending on severity)			
late submission (5%/day). submission date+time:			
late demo (10%/day). demo date+time:			
TOTAL		100	

For client/Olympus/replica functionality: Aspects not covered by a test case in the submission get at most 80% credit (penalty for inadequate testing). Aspects not covered by a test case during the demo get at most 50% credit, if the code looks reasonable.