

# Distributed Judge System for OI-Style Contests

SRIROTH Poonyapat, 1155205059

October 27, 2025

## 1 Introduction and Background

OI-style contests are a type of competitive programming competition, such as IOI and APIO. In OI-style contests, contestants are given a set of problems (typically 3-4 problems per contest). Each problem contains several subtasks, which are essentially subproblems with different constraints. Each subtask is worth a certain number of points, and a contestant's score is the sum of points from all subtasks they solve. Each subtask has different test cases, most of which are independent of each other. In this type of competition, worker judges evaluate contestants' submissions by running the code and checking whether the contestant's output matches the expected output.

The most popular judging system for OI-style contests is CMS (Contest Management System). However, CMS assigns one entire submission to a single worker for judging, which is inefficient because some workers may remain idle while others are busy. This is especially problematic for problems containing many large test cases, where one worker may take considerable time to judge a submission, resulting in slow feedback to contestants.

## 2 Objectives

This project aims to implement a distributed judge system for OI-style contests. The system will judge contestants' submissions in a distributed manner using strategies based on several factors including available workers, expected test case execution time and size, and early termination opportunities.

## 3 Proposed Approach

A master server will coordinate the entire system. It will receive submissions from contestants, compile the code, and distribute executable binaries to workers. Several distribution strategies will be implemented to efficiently utilize workers. Workers will communicate with each other to enable early termination. For example, if subtask 4 depends on subtask 3, and subtask 3 fails, then subtask 4 will not be judged. Similarly, when one test case of a subtask fails, the remaining test cases of that subtask will be skipped.