

Suggestions for CS Competitions (Task, Exam, Automatic Grading)

Wolfgang Pohl

Bundeswettbewerb Informatik

pohl@bwinf.de

Dagstuhl, January 24, 2006

Situation

conditions for IOI and similar contests

- short duration
- control (of how results are achieved)
- fairness and equality of conditions for contestants

⇒ no projects / homework tasks

exam setting,

automatic grading (trying to avoid subjectivity)

Task / Exam / Automatic Grading: Classical Task Types

- standard input/output (batch) task
- reactive task (with “adversary”, e.g. a library unit)
- output only

What else can be done – without having to invent new grading systems?

The following suggestions have not been validated.

Task Type Suggestion: Test Data

Present: a problem

Submit: test data for a given problem (input and corresponding output)

Grade (correctness): run jury solution on contestant's data and compare outputs

Grade (efficiency): run several jury solutions with different efficiency levels on contestant's data and check which solutions are made exceed limits by that data.

Similar suggestion in Cormack et al., 2006

Task Type Suggestion: Test Data / Reactive Debug

Present: a problem and a program that incorrectly or inefficiently “solves” that problem

Submit: test data for problem

Grade: run the given solution on that test data and check whether the test data detects the faults of the given solution

Instead of test data, contestants may be required to submit an adversary (e.g., a library) that detects the faults.

Task Type Suggestion: Re-Engineer Debug

Present: a program that inefficiently “solves” a problem (unknown to contestants)

Submit: a program (that should efficiently solve that problem)

Grade: run the program on jury test data (traditional)

Conclusion

Check it out!