Contest Rules

Duke Internet Programming Contest

November 17, 1992

1 Concept

The idea of this contest is to enable many people to participate in a programming contest. We hope it will be fun for all involved, as well as boosting interest in the programming teams involved in the ACM programming contests. This contest is very informal, and we hope that the contestants will compete in this spirit. The success of the contest depends on the integrity of the contestants.

2 Teams

Teams consist of *up to* four people. There are three classes of teams:

I All undergraduates or high school students.

II Mixed:

- At least two undergraduates
- No member with more than two years in grad school
- No member with a Ph.D. degree.

III Open (professors, upper-level grad students).

IV Dennis Ritchie, Don Knuth, etc.[1]

3 Contest specifics

1. We will provide a set of problems to be solved. There will be 3–10 problems in the set. These will be provided to a local contest administrator (`marc@mit.edu`).

2. The contest lasts three hours. The judges have the power to extend the contest if necessary.

3. Each team may use only one terminal (or workstation). You may use any type of computer you wish, but you must use our software to submit your solutions.

4. We will judge C programs using either the standard Sun `cc` compiler or Gnu C (`gcc`), Pascal programs using the Sun `pc` compiler, and Scheme programs using the Scheme→C `scc` compiler. The `scc` compiler follows the *Revised*^{3} *Report on the Algorithmic Language Scheme*, with minor constraints and some additions. Basically, you need to specify the "main" procedure in your Scheme program with a module specification: `(module p0 (main do-p0))`

[1] Just kidding.

would declare that the function `do-p0` is the main function in the module `p0`. The module specification should be the first line of your Scheme source code.

5. You may use any editor, debugger, and windowing system. If your program does not compile or run on our SPARCstation running SunOS 4.1.1b, that is unfortunate for you. Stick to the standard C library (similarly for Pascal and Scheme, stick to the standards).

6. No arguments will be passed to the program at run time.

7. You may work on the problems in any order, and submit them in any order.

8. You may use any books and other materials you wish, except electronic copies of pre-existing code. You may not use any code that is on-line. This rule means you must type in all code you hand in.

9. You may test problems on your computer as often as you wish.

10. To receive credit for a problem solution, you must turn it in for judging. The mechanism is described below. The response will be one of the following:

    - Correct
    - Incorrect Output
    - Incorrect Output Format
    - Incomplete Output (e.g., no output for some test cases)
    - Compiler Error
    - Runtime Error
    - Runtime Limit Exceeded (1 minute wall-clock time on the judges' machine; this should not be an issue if your program is coded correctly).
    - Submission Botch (something went wrong with how you submitted the program to us).

11. If a contestant believes that a problem is stated ambiguously, and requires clarification, he may submit a question to the judges (mechanism described below). The question and its answer may be provided to all contestants if it seems relevant. A response to the question is guaranteed, although the judges may refuse to answer any question.

12. No discussion is permitted with people outside your team.

13. All input is read from **`stdin`**, and all judged output is to be written to **`stdout`**. Output to **`stderr`** will be ignored.

14. You are responsible for backing up your own code, and for making your own printed listings.

15. Use only system include files (e.g., **`stdio.h`**), as necessary. The program should be self-contained: do not submit multiple-file solutions. The math library will be linked with your program (i.e., **`-lm`** is specified).

16. After submitting a judged run, you must wait for the response before *re*submitting the same problem. A severe penalty applies if you break this rule. You may, however, submit solutions to other problems while waiting for a response from the judges.

17. Judges may disqualify any team not competing in the spirit of the contest. Examples of unfair practices are stealing files, impeding others, degrading performance, unfair use of software tools, adjusting system clocks, etc.

18. The decision of the judges is final.

19. The judges are not responsible for network or system failures or delays.

4 Mechanisms

The mechanism for submitting judged runs and clarifications uses electronic mail. A set of programs are installed on your system that take the name of your program file, problem number, and team number, and sends the program to the judges. Your response is returned through e-mail, without code. A similar program sends a clarification request.

Clarifications and periodic scores are distributed by e-mail.

5 Scoring

- No prizes are awarded.
- The team that has had the most problems judged correct at the end of the contest will be declared the winner. There are separate rankings for each category, as well as overall. If more than one team has solved the same number of problems, the winner is the one with the fewest penalty points.
    1. Penalty points are assessed for solved problems only.
        (a) One penalty point is assessed for each minute from the beginning of the contest until the problem was solved.
        (b) The time of solution is defined to be the time of submission, according to your local computer time.
        (c) Twenty penalty points are assessed for each judged run.
        (d) Fifty penalty points are assessed for breaking rule 16.
    2. No penalty points are assessed for any problem which was not ultimately solved.
    3. The contest coordinators reserve the right to adjust penalty points to allow for unforeseen circumstances.
- The score is posted every half hour through e-mail. The scores may be withheld near the end of the contest to keep the final results a surprise.
- The final results will be mailed out (we hope) within a half-hour after the contest ends.