

2009 Call for Judges

ACM International Collegiate Programming Contest
Arab & North Africa Twelfth Regional Contest
Arab Academy for Science and Technology
Alexandria, Egypt
November 2009

The Steering Committee for the 2009/2010 ACM International Collegiate Programming Contest, Arab & North Africa Eleventh Regional Contest is seeking programming problems for the Twelfth Regional Contest to be held at Arab Academy for Science & Technology, Alexandria, Egypt in November 2009.

1.0 General Rules

1. Each contributor must submit at least two problems.
2. A contributor must have a degree in a Computer-related field.
3. Contributors must have no coaching activities with ANARC contestants during the upcoming Regional Contest.
4. The contributor of a selected problem might be given the option of becoming a judge during the contest. Judges must secure their own travel and accommodation expenses.
5. All problems must be submitted by Thursday, September 3, 2009. This date is firm and cannot be extended. Earlier submissions are encouraged.

2.0 Guide for Problem Contributors

Each contributor must submit at least two problems, each consisting of:

1. A problem statement (description);
2. A solution in ANSI C, C++, or Java; And
3. Input/Output files to use for judging.

2.1: Problem Statements

1. Problems must be original. Do not send past problems, problems taken from the Internet or text books, problems you've used in class work, training sessions, etc.
2. Problem statements should be written in unambiguous, simple English. English is not the first language for most contestants.
3. All problems must require input.
4. Unless the core of the problem is input/output related, the formats chosen for input data and the displayed results should be relatively simple. Still, the format of the input data and the appearance of the expected displayed results must be described in suitable detail.
5. Input must be taken from a single input file. Nevertheless, multiple data sets testing different cases are appropriate; make the problem statement include iterative data sets. See past problems sets for examples.

6. Anticipate questions about special cases. Where appropriate, explicitly state that certain special cases will not appear in the input data. It is not necessary to specifically identify the special cases that will appear.
7. Contestants must write solutions for problems in a short time. While very simple problems are not appropriate, neither are problems that require a great deal of code; a few hundred lines of Java or C should be an upper limit on what can be expected in a solution.
8. The program and chosen test data should not require excessive execution time. Contestants' solutions may be less efficient than yours and so a generous margin is allowed for execution. Your solution should solve all test cases within 10 seconds on a moderately equipped computer.
9. The problem description (excluding sample input/output) should fit on a single A4 page using a 12pt font. Exceptions are allowed if the problem statement include big figures. Check past problem sets.

2.2: Judges' Solutions

1. For each problem you propose, prepare a solution in C/C++ or Java.
2. Include comments in your code, even though the contestants' code need not be commented.
3. Make sure that your program correctly solves the problem! Include test data that illustrates the generic and special cases that you expect the contestants' solutions to handle.

2.3: Test Data

1. Data must be unambiguous when printed. Consider carefully those cases where trailing blanks (or leading blanks, etc.) will make a difference in a program that processes input data.
2. If several test cases are included, describe the manner in which data for the test cases is separated in a single file.
3. Include a rationale for each of the test cases you provide. This will help identify missing test cases as well as identify those cases where a student solution fails.
4. Put a copy of the sample input data first followed by general cases, ones which student solutions are likely to get. Stress tests (boundary values) should appear last.

3.0: Submission of Problems, Solutions, and Test Data

1. Use a separate directory for each problem. Pack all the files in a single zip file.
2. Problem description can be sent as Adobe PDF, Plain Text, or, as a last resort, MS Word document.
3. EMail the zip file to rcd@icpc-anarc.org. Please indicate if you're willing to participate as a judge if some of your problems get selected.

4.0: Useful Links

Check the training page [↗₂](#) for repositories of previous ICPC and ANARC problems.

(Subscribe to the Announce Mailing List [↗₃](#) if you wish to receive future call-for-judges by email.)

Ziad H. Najem, PhD
2009 ICPC/ANARC Regional Contest Director
Kuwait University
email: rcd@icpc-anarc.org
Tel. +965 99 68 31 52

URL Links

- [↗₁ http://www.acm-acpc.org](http://www.acm-acpc.org)
- [↗₂ http://www.acm-acpc.org/direct/training](http://www.acm-acpc.org/direct/training)
- [↗₃ http://mailman.icpc-anarc.org/listinfo/announce](http://mailman.icpc-anarc.org/listinfo/announce)

About This Document

This document is part of *The ACM/ICPC Arab Collegiate Programming Contest* website, <http://www.acm-acpc.org> [↗₁](#). Send queries and comments to info@acm-acpc.org.

Copyright © 2007-10 ACM/ICPC Arab Collegiate Programming Contest.