Computer Science Competition

Region Contest Director's Guide

The UIL Computer Science Competition consists of two components – a multiple choice exam and a hands-on programming session. The same students from a school participate in both components. **All contestants must take the written test.** If a team has four members, the team's coach must select three of the four to participate in hands-on programming. A team must ***both*** have all team members take the written test ***and*** have three participants in programming in order to be eligible for team competition.

The Written Test

All contestants must take the multiple choice exam (the "written test"). Individual scores, placement and advancement are determined solely by scores on the written test.

Conducting the written test is straightforward. The test consists of 40 questions – the first 38 are multiple choice, and the last two questions are free response. (Free response questions are designed to have one discrete answer). Contestants have 45 minutes for the test, exclusive of time required for instructions. Contest directors may use the answer sheets provided in the contest packet, or they may choose to use Scantron grading.

You will receive from the UIL office a packet of materials for conducting the written test – this packet will include tests, contestant answer sheets and answer keys for grading.

To assist you with tabulating individual scores and tiebreakers, an Excel scoring calculator for the written test is included with your judging materials. ***Use of this spreadsheet is optional and does not take the place of entry in the online system – all individual scores must be entered in the online system prior to verification.***

Programming

Advancing teams must also participate in the programming session in order to be eligible for team placement and advancement. Written exam and programming session scores are combined to determine overall team scores.

The remaining pages of this document describe how to prepare for and run the programming session. You will receive from the UIL office judging materials, as well as a packet of programming problem sets (the tests).

UIL Computer Science
Region Programming Contest – Director's Guide

Regional Computer Science Contest Directors
- The Computer Science Contest Director (CSCD) is a designated UIL official, and as such has the authority to make decisions regarding the management of the contest to ensure that it is conducted according to UIL rules.
- The Computer Science Contest Director also has the authority to settle any disputes that might arise, in accordance with UIL rules.
- If a situation should arise that you are not sure how to resolve, or if you have any questions regarding rules, procedures, etc., please feel free to contact David Trussell, UIL Director of STEM Activities, at the UIL state office: dtrussell@uiltexas.org, or 512-232-4926.

Fundamental Contest Rules
- **Teams:** For the Programming component, three team members participate. If a team has four members the coach decides which three members participate in Programming. There must be at least 3 contestants to participate in team competition.
- **Hardware:** Each team has one computer consisting of a single monitor, keyboard, and system. If laptops with external keyboards are used, the laptop keyboard must be covered in some way (for example, taping a piece of paper over the keyboard). A team may use one printer, but a printer is not required.
- **Software:** The computer may be loaded with normal software (such as word processors, etc.), the Java JDK, and a development environment (such as JCreator, Eclipse, NetBeans, or whatever IDE the team chooses).
- **Media:** Most regional sites are running networked contests using PC^2 software (see next page). For non-networked contests the preferred media format is the USB flash drive. Teams should have at least 3-4 flash drives so they can submit more than one solution at a time.
- **Allowed resources:** The documentation for the Java API's, and the API's themselves, may be installed on a team's computer. Teams may also have up to two hard copy published textbooks or reference books that are reasonably free of handwritten notes.
- **Non-allowed resources:** It is important that the team's computer system be free of any code written by the contestants prior to the contest such as solutions to practice problems, solutions to prior contest problems, or programs written in class. A book on a CD or on a team's hard drive is not an acceptable reference and should be removed from the computer system.

Contest Materials from UIL – Inventory all items, make any necessary additional copies and keep secure until contest day.
- Judging Materials – UIL will provide download access to judging data files, contestant sample data files and options for judging software. Materials for download also include instructions and original documents to be copied for precontest team packets. Make one packet for each team prior to contest day. Judging software options included are PC^2 for networked contests and a DOS-based judging environment for non-networked contests.
- Printed Problem Sets (the tests) – distribute one copy to each team at beginning of contest. Remaining copies are for judges' reference. ***After inventory, keep this packet sealed until contest.***
- Printed Judge Packets – copies of printouts of the judge data files, to be used for reference during judging. Keep these packets secure until the contest, and provide copies only to judges during the contest. ***Do not distribute to contestants.***

Networked Programming Sessions – An Overview of PC²

All regional sites are strongly encouraged to adopt a networked contest format for the programming session. PC² is the easiest platform to use for this purpose. Contest directors will need a standalone wireless router to set up a contest network (NOT Internet connected) and a computer to use as the contest server (a reasonably current laptop is fine), along with the standard 2-3 judging stations. The software requires a little time for becoming familiar with the setup and configuration procedures, but it is time well spent – networked contests run much more smoothly and efficiently than the old non-networked "sneaker net" format.

Below is an overview of PC², with information on where to find complete documentation and download the software. In addition, UIL offers customized extensions for the software, as well as setup/configuration support for all regional contest directors – information will be distributed as the contest dates approach.

PC² (the *P*rogramming *C*ontest *C*ontrol system, pronounced "P-C-squared" or sometimes just "P-C-Two" for short) is a software system designed to support programming contest operations in a variety of computing environments. PC² allows contestants (teams) to submit programs over a network to contest judges. The judges can recompile the submitted program, execute it, view the source code and/or execution results, and send a response back to the team. The system also supports an "automated judging" mode where judging is performed by software rather than by human judges. *(Use of automated judging is NOT recommended for UIL regional contests.)*

The system automatically timestamps and archives submitted runs, maintains and displays current contest standings in a variety of ways, and allows the judges to retrieve and reexecute archived runs. It also provides a mechanism for contestants to submit clarification requests and queries to the judges, and for the judges to reply to queries and to issue broadcast bulletins to teams.

A wide variety of configurable options allow the contest administrator to tailor the system to specific contest operations. For example, the number of teams, problems, and languages in the contest; the scoring method being applied; which problems are handled by which judges; whether teams are automatically notified of the result of a submission; and the frequency of automatic scoreboard updates are all configurable. There are also mechanisms provided for editing the internal scoring database, and for recovering from various types of soft and hard errors. The system is designed to allow teams to use any language development tool which can be invoked from a command line and generates an executable file.

PC² was developed at California State University, Sacramento (*CSUS*), and is available on the World Wide Web at https://pc2ccs.github.io.

Computer Science Contest Director's Guide – Duties and Procedures

The following is a chronological list of duties for the Computer Science Contest Director. *Note that specific procedures will vary somewhat depending on the judging platform used, particularly if running a networked contest using PC^2.*

Computer Science Contest Director (CSCD) Duties - before the contest day

- Appoint additional personnel to assist in running the region Programming contest. For a 9 team region the CSCD will need:
  - 2 or 3 assistant judges - The CSCD along with the additional judges will judge solutions submitted by the contestants. Coaches from participating schools may serve as judges.
  - Appoint 1-2 room monitors (Coaches or willing teachers are good room monitors)
    - The room monitor(s) will assist with checking teams in and then seat them.
    - The room monitor(s) will circulate in the room to see that all teams are adhering to contest rules both before and during the contest.
    - The room monitor(s) will supervise the runners when they are in the contest area.
  - For non-networked contests, appoint 1-2 runners (to be supervised by the room monitor).
    - The runners pick up solutions from the contestants and take them to the judging area to be archived, judged, and scored.
    - The runners also return judged submissions to the teams.
  - For non-networked contests, appoint a scorekeeper and archivist - These may be separate or combined positions or the CSCD and the assistant may do this depending on the number of competing teams.
    - The archivist backs up solutions submitted by teams prior to them being judged.
    - The scorekeeper records results of judged solutions using the provided scoring program.
- See that necessary software is installed on the judging stations (current Java JDK version required)
  It is strongly recommended you install and verify that contest and judging software works a few days prior to the day of the contest. This will allow you enough time to resolve any problems you may encounter.
  - If applicable, install the scoring program for the Scorekeeper.
  - If applicable, set up archiving directories for the Archivist.
  - Install the judging data on the judging stations. (If you use the DOS-based judging environment, the judging data is installed automatically.)
  - Test the judging software by running the solution to the dry run. A solution to the dry run is provided with the judging materials.
  - If applicable, test the Scorekeeper's software, and the Archivist's station.
  - If you wish to use the provided scoring program, install that software on the computers that will be used by the archivist / scorekeeper. (Refer to instructions later in this document).
- **Create sample data distribution for contestants.** A folder containing the contestant data files is included with the Judging files. Copy the student sample data files from the judging materials onto media for distribution – provide one copy for each team. **Include the student data files ONLY.** (Flash drives work well for this purpose – you may choose to collect a flash drive from each team prior to the contest, copy the data files, then return the drive just before the contest begins when the problem sets are handed out.)

CS Contest Director Duties (contest day)
- Obtain signs from the Meet Director and post them to identify each station and team area.
- Ensure judging, archivist and scorekeeper stations are set up.
- Ensure Precontest Materials are ready for distribution as students check in.
- Ensure Official Contest Materials are ready for distribution at the beginning of the contest.
- Train judges, archivist, and scorekeeper on the software and their duties, as applicable.
- Train runners on their duties (if applicable).
- Ensure Judges materials are ready for distribution when the contest begins.
- Give name tags to contest personnel.

CS Contest Director Duties (at least one hour prior to the start of contest)

- Check-in: CSCD or an assistant will check teams in and confirm the members' names.
- Team Members: In the Programming session, three team members participate. If a team has four members the coach decides which three members participate in the Programming competition. *There must be 3 members present and participating in the Programming session to be eligible for team competition. All Programming participants must also take the written test.*
- Precontest Materials: Give the teams their Precontest Materials packet as they check-in.
  - o Precontest team packets – One manila envelope per team, each clearly marked "Precontest Materials" containing (original documents for copying are included with judging materials):
    - ▪ One copy of the Programming Session Instructions page.
    - ▪ One copy of the dry run problem (dry run is problem number 0).
    - ▪ One Team Verification sheet (colored paper).
    - ▪ Scratch paper – include a few sheets of blank copy paper
    - ▪ You may wish to provide each team with a copy of the data file for the dry run problem.
    - *~ The three items below are not needed for networked contests using PC^2. ~*
    - ▪ 3 or more manila run envelopes (9x12).
    - ▪ 13 run sheets.
    - ▪ 4 Clarification Request sheets. (colored paper preferred).
- Dry run: At check-in, explain to each team the process for submitting their dry run solutions.
  - o The dry run problem is a simple problem to ensure understanding of the contest operations and to be sure the contestants, the runners, archivist, judges and scorekeeper are all working together.
  - o The dry run problem is included in the Team Precontest Materials packet. Teams may have a working version of the dry run solution prior to coming to the contest. If not, they should write the solution as soon as they have their system set up.
  - o **NOTE:** For networked contests using PC^2 software, the processes described below will all take place over the network – the software handles archiving and scorekeeping functions, as well as transmitting solutions to judges and judge responses back to the teams.
  - o After the teams have their computer system set up, they are to submit their solution over the network, or place the source code for their dry run solution on a flash drive and insert the drive and completed run sheet into a run envelope and give it to a runner when told by the CSCD.
  - o The runner will take the run envelope to the archivist, who will archive it and give it to the judge to be judged.
  - o The judge will compile and run the contestant's solution using the judging data. Expected results are compared to the actual results from the contestant's solution.
  - o The judges will then record on the run sheet if the solution is accepted or rejected and indicate the appropriate response on the run sheet. Then place the flash drive and run sheet into the run envelope. Give the run envelope to the scorekeeper.
  - o The scorekeeper will record the results and give the run envelope to the runner who will return it to the contestants.
  - o Teams may continue to submit solutions to the dry run problem until they get it correct or until the time for submitting the dry run is over. *It is NOT required that each team have a correct dry run solution before continuing with the contest.*
  - o Contestants may confer with their coach in the contest room during the dry run process.
- Monitoring: The CSCD or an assistant will monitor the contest room as teams set up their computer systems and complete their dry run.

CS Contest Director Duties (30 minutes prior to the start of contest)

- Ensure teams have a single computer system.
  - o Each team is allowed one computer consisting of a single monitor, keyboard, and system. If laptops with external keyboards are used, the laptop's built-in keyboard must be covered in some way (for example, a piece of paper may be taped over the keyboard). Teams may use one printer, but a printer is not required. The team may have a back-up computer system provided it is packed and stored in a nearby area (e.g. front of room) unless actually needed.

- Ensure all previously written programs have been removed from the contestants' computer systems.
  - o Allowed Software: The computer may be loaded with normal software (such as word processors, etc.), the Java SDK, a development environment (such as JCreator, Eclipse, NetBeans, or whatever IDE the team chooses, including any sample files installed with the software).
  - o Allowed resources: The documentation for the Java API's, and the API's themselves, may be installed on a team's computer. Teams may also have two hard copy published textbooks or reference books that are reasonably free of written notes.
  - o Non-allowed resources: It is important that the computer system be free of any code written by the contestants or coach prior to the contest such as solutions to practice problems, solutions to prior contest problems, or programs written in class. A book on a CD or on a team's hard drive is not an acceptable reference and should be removed from the computer system.

CS Contest Director Duties (Conducting the Contest)

- Assemble contestants and coaches for instructions. Go over contest rules on the front cover of the contest problems and any pertinent procedures from the UIL Computer Science Handbook.
  - o There are 12 problems on the test.
  - o Explain the scoring.
    - Each correct solution will score 60 points.
    - Each incorrect solution will score –5 points **only if** a correct solution is eventually submitted.
  - o The contest will last for 120 minutes.
  - o Teams may work on the problems in any order.
  - o Go over the contest mechanics. Describe the process for submitting solutions over the network when using PC^2, or for non-networked contests describe the process below:
    - Complete a run sheet with their team number and problem number on it
    - Save the source code (.java file) to a flash drive
    - Flash drives should be labeled with a team name or number if possible.
    - Place both the run sheet and the flash drive in the run envelope for that problem.
    - Hold the envelope HIGH in the air for a runner to collect.
    - The runner will return the problem as soon as it is judged and recorded.
  - o Clarification Request form – For networked contests, clarification requests are also handled through the PC^2 software. Clarification requests are used to pass information between a team and the judges. For non-networked contests, the Clarification Request form is filled out by a team and given to a runner to take to the contest director. The purpose of the form is to resolve ambiguities in a problem statement. **Clarifications are not to be used to give teams an advantage, extra information, or hints on how to solve a problem.** The contest director must decide on the appropriate response in consultation with the other judges. It may well be that the appropriate response is for the team to read the problem statement more carefully and look at the given examples. If a clarification is in fact required all teams should be made aware of the resolution.
  - o **The judging data will include test cases not shown on the problem statement sheet**.
  - o When the 120 minutes is over, you will say "Stop". For networked contests, stop the contest on the PC^2 server software. For non-networked contests, only accept remaining solutions that are already in a run envelope, with the run envelope in the air.
  - o When the contest is over, teams will complete their Team Verification form. Each team will then give their Verification sheet to a contest official who will take it to the judging room.
  - o Students may not disassemble their computers until all judging is complete and all questions have been resolved. Once a team disassembles their computer system, they forfeit their right to further questions on problems or judging.
- Instruct the teams to delete their dry run solution from their computers.
- Answer any final questions.
- Ask the coaches that are not helping to run the contest to leave the room or be room monitors.
- Distribute Official Contest Materials (Problem Sets and contestant sample data). Instruct students not to open them or insert the sample data flash drives until they are told to "Start".
- Set the timer to 120 minutes. Announce "Start" and start the timer or the PC^2 contest server. This is the official start of the contest. Contestants may open their packet and have two hours to submit solutions. All solutions must be submitted using the procedures described for a networked or non-networked contest.

**NOTE: The process below describes a non-networked contest. For networked contests, the PC²
software will perform many of these functions. In either case, TAKE CAREFUL NOTE OF THE
SECTION ON JUDGING GUIDELINES.**

- **Archivist Duties:**
  - Records the team number and problem number for the submission.
  - Records the time the problem was submitted.
  - Backs up the solution submitted.
  - Gives the run envelope to a judge.
- **Judges Duties:** Submissions should be maintained in first in first out order. As judges become available they should judge the oldest submission.
  - If using the DOS-based judging environment provided, follow the instructions in the "Running the Judging Environment" document.
  - If not using the judging environment:
    - Create a folder for each team.
    - Copy the file to the team's folder.
    - Remove the disk from your computer.
    - Compile and run the program submitted.
  - **IN EITHER CASE, USE THESE JUDGING GUIDELINES:**
    - White space differences at the end of lines or after the last line of output are never significant and should never cause a solution to be judged as incorrect.
    - If the differences do not seem material to the problem being solved, err on the side of accepting the solution. For instance, if a problem is about performing a complex calculation, be flexible with output formatting. On the other hand, if the problem is all about formatting, then be a stickler.
    - **Above all, be consistent with your judging.**
  - If a submission is correct, mark accept on the run sheet.
  - If a submission is incorrect, mark reject on the run sheet and mark one of the comments on the run sheet. State judges generally only used a small number of the available comments:
    - "Does Not Compile" is used for submissions that suffer compile errors.
    - "Run-time Error" is used for submissions that suffer a runtime error or exception.
    - "Failed Judges' Test Data" is used for any cases where the submission compiles and runs but whose output does not match the expected answers.
    - "Time-Limit Exceeded" is used when a submission results in an infinite loop or the program does not end. None of the problems in the problem set require a large amount of computation. A general guideline for when to halt a team's program is 2 minutes.
    - After judging the solution and marking the run sheet, the submission disk and run sheet go back into the run envelope which is given to the Scorekeeper.
- **Scorekeeper Duties:** The scorekeeper records successes and failures of the contestant's solution using the provided scoring program and then returns the run envelope to the runner.
  - Correct solutions are awarded 60 points less any penalty points.
  - Incorrect solutions are penalized 5 points for EACH time an incorrect solution is submitted ONLY WHEN the correct solution is finally submitted. For example:
    - Team A submits two incorrect solutions for problem 3 and never submits a correct solution. Team A will receive no points and will be assessed no penalty points for problem 3.
    - Team B submits two incorrect solutions for problem 4 and then submits a correct solution, Team B will receive 50 points for problem 4. This is 60 points for submitting the correct solution minus 5 points for EACH incorrect submission (total of 10 penalty points).
- The runner returns the run envelope to the team.
- If a team's solution is judged incorrect, the team may rework their solution and resubmit their new solution. Teams may make as many submissions for a given problem as they wish.
- All teams should be notified when 15 minutes are remaining in the contest.

**Note: Coaches may fill any of the above positions, and one person may serve in more than
one of these positions.**

CS Contest Director Duties (Ending the Contest)

- At the 120 minute mark, teams shall be told to stop.
- Stop the contest on the PC^2 server, or for non-networked contests, accept any solutions that are in a run envelope and being held up in the air when the signal to stop was given.
- No further submissions are to be accepted.
- Tell students to leave their computers set up until judging is complete. Failure to do this will cause a team to forfeit their right to further questions on problems or judging.
- Have teams complete their Team Verification forms and turn them in to a contest official.
- Complete judging.
- Complete scoring. The maximum possible programming score is 720. Ties are not broken. (Overall team ties are broken when written exam and programming scores are entered in the Spring Meet Online System).
- **VERIFICATION PERIOD:**
  - Allow a period of 15 minutes immediately following the contest for teams to ask questions.
  - Establish an orderly process for conducting verification that works efficiently for your number of teams and contest personnel (teams line up at the front of the room, remain in their seats and raise their hand, etc.).
  - The purpose of the verification is for teams to ask specific questions about specific problems where they believe they should have received credit. It is not a time for more general inquiries about how to solve a problem. It is important to keep the process moving so questions can be addressed and the contest can be concluded in timely fashion. Once the 15 minutes has elapsed, close verification and wrap up any questions that are in process.
  - At the discretion of the contest director, it is permissible to show teams how their program runs on the judging station with the judge test data.
  - DO NOT share with teams the sample solutions provided with the judging materials during verification. These samples are intended to be released to teams after the meet to provide an example of how each problem can be solved.
  - Before or during the verification period described above, also verify recorded scores with Team Verification forms and resolve any discrepancies.
- For Friday contests, collect all official contest materials following verification. These may be returned to teams when other official materials are returned. For Saturday contests, materials may be returned after verification is complete and official results are announced.
- **DATA ENTRY: It is essential to complete the data entry process prior to the announcement of official results.** (The online system will help you avoid any mathematical errors in the tabulation of individual and team scores, as well as handling ties if needed.) Work with your meet director to ensure that your data entry can be completed in timely fashion.
  - If your written exam happens first, enter ALL contestant scores prior to the verification period for the written exam. The system will allow you to print a list of individual scores and places (unofficial results) that can be used during verification. Complete verification for the written exam, resolve any issues, then declare the individual part of the contest official. After the programming session, follow the verification procedure described above. Once that is complete, enter programming scores in the online system. The system will calculate total team scores. Announce unofficial results and allow a few minutes for any questions about data entry of programming scores. Resolve any issues, then declare the team part of the contest official.
  - If your programming session happens first, complete the programming verification procedure described above. Keep final programming scores in a safe location so they can be accessed for data entry. After the written exam, enter ALL contestant scores prior to the verification period. Also enter programming scores at this time. The system will allow you to print a list of scores and places (unofficial results) that can be used during verification. Complete verification for the written exam, along with verification of final team scores. Resolve any issues, then declare contest results official.

Thank you for your willingness to host the UIL Computer Science contest at the region level and for all of your hard work!