



University Interscholastic League
Computer Science

Contest Introduction

Dr. Shyamal Mitra

State Contest Advisor

For new coaches and contestants.



In This Session

- Contest introduction, including
 - general rules
 - contest structure and scoring
 - the hands-on contest
- State written test
- Sample hands on questions



Contest Directors & Test Writers

- Sam Armstrong
- Rich Brozovic
- Scotty Johnson
- Matthew Sheldon
- Grant Wilde
- Dr. Shyamal Mitra
- David Trussell



What is the Computer Science Contest?

- A competition that challenges students to apply computing and algorithmic concepts and skills
- Tests knowledge of algorithms, computation, and object-oriented programming
 - using the Java programming language
- Allow students to expand their knowledge of computer science beyond what they learn in the classroom and to foster their interest in the field



Background

- Contest established in the 1990 – 91 school year
- Modeled on the College Board's Advanced Placement Curriculum for computer science
 - plus some additional topics
- Brought computers into UIL competition for the first time with State Meet hands-on programming
 - Modeled on ACM programming contest
 - Programming was later added to regional and district and is also now included with UIL invitational materials



Programming Language

- UIL - same language as the AP curriculum
 - if AP changes, we will too
- Pascal for 8 years. C++ for 5 years. And now more than 20 years in Java.
 - language just a tool to test concepts
 - example: sorts are essentially the same
- What version of Java?
 - the most recent version as of September of each school year



General Rules

- As with other UIL academic contests
 - Participants must meet eligibility requirements
 - A school may enter up to six contestants in the district competition



Contest Structure

- Two components: individual and team
 - The same contestants from a school compete in both components
- District, Regional and State competitions consist of:
 - a 45 minute, 40-question written exam, for the individual competition and the team competition (counts for half of team score)
 - a two hour, 12-problem hands on programming contest for teams



The Individual Component

- At all levels of competition, individual places are determined solely by written exam scores.
- All contestants compete for individual honors at all levels of competition
- Individuals placing first, second, and third advance to the next level of competition



The Team Component

- ALL team members **MUST** take the written exam
 - Three members participate in programming
- At all levels of competition team placement is determined as follows:
 - top three team member written scores + programming score = overall team score
- First-place and wild card teams advance to the next level of competition



Team Entries and Scoring

- A school may enter up to six contestants at district, or a district may vote to allow only four contestants per school
 - In districts with six contestants per school, the top four scoring individuals on the written exam will constitute the school's team
 - All four members of first place teams and wild card teams advance to the next level of competition
- A school must enter at least three contestants to participate in the team competition
- The top three written exam scores from a school are counted towards the team score plus the score from programming
- The same contestants compete for both individual and team awards



Participation Requirements

- ALL contestants **MUST** take the written exam at all levels of competition
- Teams **MUST** participate in programming in order to qualify for team placement or advancement
 - Exception – if only one team is entered in the district contest, conducting the programming session is encouraged but not required (if two or more are entered, programming is mandatory)



Scoring Rules – Written Exam

- 40 questions
- SIX points awarded for correct answer
- TWO point deduction for each incorrect
- No points given or deducted for unanswered questions
- Questions may be skipped
- A 15 minute verification period is held prior to announcing official results
- Verification is your chance to ensure that grading and tabulation are correct



Scoring Rules – Programming

- 12 programming problems
- 60 points awarded for a correct answer
- 5 points subtracted for each incorrect answer only if a team eventually gets a correct answer
- Incorrect solutions will be returned and may be reworked and resubmitted
 - judges do not provide a detailed explanation of why a solution is incorrect



What About Ties?

- In individual competition ties are broken by determining the highest percentage of correct answers
 - Example:
 - attempting 30 questions with 20 correct
 $= 20 * 6 - 10 * 2 = 100$
percent correct = $20 / 30 = 66.7\%$
 - attempting 22 questions with 18 correct
 $= 18 * 6 - 4 * 2 = 100$
percent correct = $18 / 22 = 81.8\%$ (wins the tie break)
- If a tie still exists it will not be broken



Ties, continued

- In team competition, ties are broken by the team that has a higher score on the programming portion
 - if a tie still exists the total team score on the written exam is considered
 - if a tie still exists it will not be broken



Wild Cards

- The highest scoring second place team among all districts in a given region advances to the regional meet
 - one wild card per region
- The highest scoring second place team among all regions advances to state
 - one wild card per conference
- Districts must report their team scores with contest results on time to be eligible for the wild card



Written Contest Materials

- Pencils and erasers
- Scratch paper is provided
- No calculators



Written Contest Format

- A 45-minute exam consisting of 40 questions
 - Questions 1-38 are multiple choice
 - Last two questions are free response
 - These will have discrete short answers
- Answers are recorded on the answer sheet
- Topic list provides areas covered
 - Specifies topics for first 15 questions
- Old exams are very useful for practice



Programming - Materials

- In programming three members of a team participate
 - coach's choice
- Each team may bring two published reference texts
 - includes textbooks and language manuals
 - books should be reasonably free of written notes
- For “sneaker net” contests, each team must bring media for submitting solutions to judges, typically USB flash drives.
 - Bring several flash drives (at least 3 or 4), since some drives may still be in the judging room when a team has another solution ready to submit
 - Smaller capacity drives are fine – better to have more drives than larger capacity



Programming - Computers

- Each team shall be prepared to bring one computer to use for competitions
 - some sites may provide computers but check with local contest director
 - most district sites, regional sites and state require teams to bring their own computers
 - Mac users may need to bring an additional computer for the judging station
- printers are allowed, but not required



Programming - Computers

- Each team may use **ONLY ONE** computer
 - one monitor, one keyboard, one mouse
 - no dual monitor or dual keyboard/mouse systems
 - you can bring a backup computer



Programming - Computers

- What software can be on the computer
 - operating system
 - standard software preloaded on new computers: office, explorer, anti-virus
 - A Java compiler and IDE
 - Built in libraries, library documentation, and help functions may be used during the contest



Programming - Computers

- What CANNOT be on your drives:
 - Solutions, data files, templates, from previous UIL competitions or any other programming competitions
 - Programs written for class
 - Any other program written by contestants or coaches



Programming - Judging

- Computer setup for judging will vary from site to site
 - Most sites will have judging stations in a room separate from the contest room
 - other arrangements possible
 - Most regional sites and the state contest use a networked contest system
- Check with your host site ahead of time to find out what procedures will be used



Programming - Judging

- Contestants submit Java source code
- Judges recompile and run on test cases
- No major problems with using Java thus far



Programming Contest Format

- A two-hour programming contest consisting of 12 problems
 - varying degrees of difficulty, but all worth 60 points
 - finding the easy ones is part of the competition
- Plan to arrive early to allow time to set up equipment and have systems verified
- Prior to the beginning of the contest teams will work a simple dry run problem
 - a system check for contestants and judges
 - not necessary for all teams to complete the dry run successfully before beginning the contest



Programming Contest Format

- Typically, contestants work in one room while judges work in another nearby room
- Teams submit solutions as they finish them over the contest network or on a flash drive (along with a run sheet)
 - for “sneaker net” contests, runners transport materials between contestants and judges
- For a correct solution, judges return an acceptance notice over the network, or return the flash drive and an acceptance form



Programming Contest Format

- When a team submits an incorrect solution, the judges respond over the network or return the flash drive and run sheet
 - general comment on problem
 - syntax error
 - runtime error
 - failed test case
 - exceeded time limit
 - NO information on why solution is incorrect
 - teams may rework the solution and resubmit it



Programming Contest Format

- Teams can submit a clarification request if they believe the problem is unclear
 - many times the response from judges will be to read the question more carefully
 - judges will not explain unfamiliar concepts during the competition
- Standings may be posted periodically during the course of the contest



Programming Contest Strategy

- Break up the problem pack
- Find the easy problems
- One person working on an easy problem on computer
- Two others working other problems on paper
- Problems may be worked in any order
- Know when to give up on a problem
 - computer time is a scarce resource



Returning Papers

- If there are no unresolved questions then at the district level entries may be returned no sooner than the end of the last contest day of the district week
- If there are no unresolved questions then at the regional level entries may be returned to contestants at the end of the day on Saturday of region weekend



Frequently Asked Questions

- Can team contestants receive individual awards if they did not place in the individual competition at the previous level competition?
 - Yes. Team contestants are in the mix for individual honors, even if they did not place in the top three at the previous level of competition



Frequently Asked Questions

- Do contestants who advance only as individuals participate in programming?
 - No. Contestants who advance as individuals only take the written test at the next level of competition



Frequently Asked Questions

- If a team gets a solution correct on the second or third or later try, do they still receive the 5-point deduction?
 - Yes



Frequently Asked Questions

- What if one of our team members is sick or otherwise unable to compete at regionals or state? May we substitute?
 - Yes. Advancing teams may insert a substitute for one and ONLY ONE team member who is unable to compete at the next level of competition.
 - If more than one member is unable to compete the alternate team will advance
- Can substitutes win individual awards?
 - Yes



Preparing for The Contest – Online Resources

- UIL
 - www.uiltexas.org/academics/stem/computer-science
- Includes resource page with links to
 - java compiler and IDEs
 - second party materials
 - references
 - online programming problems



Preparing for the Contest - Books

- Building Java Programs
 - Stuart Reges & Marty Stepp, Pearson, www.pearson.com
- Introduction to Java Programming
 - Y. Daniel Liang, Pearson, www.pearson.com
- Java Language Specification
 - <https://docs.oracle.com/javase/specs/>
- Your classroom textbook



Preparing for the Contest Development Tools

- IDE (interactive development environments) are tools that allow you to write Java programs
- You don't have to use one
- You can use whichever one you want
- Demos of
 - command line
 - textpad
 - Eclipse
 - BlueJ



IDE Information

- Eclipse
 - <http://www.eclipse.org>
- BlueJ
 - <http://www.bluej.org/>



Preparing for the Contest

Practice problems

- Coding Bat
 - <http://codingbat.com/>
- Kattis
 - <https://open.kattis.com/>
- Programming Challenges
 - <https://uva.onlinejudge.org/>
 - online problems and judge



Preparing for the Contest

Practice problems

- USA Computing Olympiad
 - <http://www.usaco.org/>
- Project Euler
 - <https://projecteuler.net/>



Practice Test 2023

- Review Questions
- Reference Sheet
 - use this to help answer questions
- Topics List
 - check the CS page of the UIL website for the final 2024-2025 list



Questions

- www.uiltexas.org/academics/stem/computer-science