Preparing a Team for the
ACM Scholastic Programming Contest

MODERATOR

Donald Bagert, Texas Tech University

PARTICIPANTS

Darrah Chavey, Beloit College
Thomas L. Mourey, Mesa State College
David Van Brackle, University of Central Florida
John Werth, University of Texas at Austin

ABSTRACT

Every year, more universities are increasing the amount of preparation given to teams competing in the ACM Scholastic Programming Contest. This panel will have team faculty advisors (i.e. *coaches*) of teams which have reached the 1991 Contest Finals discuss some of the techniques they use in preparing their students for regional and international competition. Some specific issues to be addressed are: What kinds of problem solving strategies should be covered in practice? How important is the concept of "scrimmaging"; i.e. holding a simulated contest using previous contest problems? Is there a better "throughput" of problems solved by working on the problems individually, or in teams of two or more?

STATEMENTS

Darrah Chavey

Scrimmaging is a necessary preparation for the Programming Competition, but students should be prepared for the scrimmages. This preparation should include: techniques for selecting problems; techniques for attacking problems; and techniques for working together.

Thomas L. Mourey

While explicit practice on recognizing different problems is important - even necessary, nothing can take the place of practice in a contest environment. This means scrimmaging against other students from our school (the second team for instance), or better, against other schools. Despite our isolated circumstance (the nearest school is 150 miles away), we try to have inter-school contests as often as possible - usually about 2-3 times each year.

David Van Brackle

UCF's teams are chosen at a local contest. Students compete as individuals. The top 9 form 2 teams with an alternate. The top 4 aren't necessarily the 1st team; rather, the 4 are chosen who have the best combination of personalities and skills. The teams undergo a rigorous training schedule. Each week, they compete in a 6-hour "mock" contest using problems from previous contests and problems supplied by faculty and other volunteers. We feel that these mocks are necessary to prepare for a 6-hour contest. For 1 hour before and after each mock, we discuss strategy. These discussions cover a wide range of topics, including allocation of the terminal time to people, allocation of problems to people, and problem-solving and algorithmic strategies. The emphasis is largely on team dynamics, taking 4 good programmers and teaching them to work together as a good team. There are many specific strategies of team organization and dynamics which we work on in the mocks, too many to be listed here. However, there are 2 fundamental tenets: do the easiest problems first, and do as much work away from the terminal as possible. Each programmer on our teams works as an individual. We feel that this maximizes throughput. There are times when they work together - when there are less than 4 problems left, or when a problem requires the specific skills of more than 1 of them, but for the most part, the 'parallel processing' of 4 programmers working on 4 programs gets more problems solved more quickly.

John Werth

Picking and training teams requires attention to the abilities needed for success and the key events in the contest. The abilities, in no particular order, are knowledge, programming skills, strength (five hours is a long time), appropriate team skills, and competitiveness. The key events are arriving ready to compete, picking the right problems to solve, assigning the problems to the right people, testing before submitting, sharing the machine, and the endgame. To pick a team, I favor giving a local contest for small groups that is similar to the national one in difficulty and effort. Then combine the small groups into teams. The key is to get the best people to compete in this local contest. To train a team, focus on the abilities and the key events described above. I strongly prefer an amateur approach to the entire process. This means participation for the "fun of it" by students and faculty. It also means not paying students, whether with academic credit or scholarships, for participating or training.