"FIT": FIELD EXPERIMENTAL EVIDENCE ON CREATIVE WORKER SORTING ON AN INNOVATION TASK

Kevin J. Boudreau, London Business School, kboudreau@london.edu
Karim R. Lakhani, Harvard Business School, k@hbs.edu

Abstract:

We present the results of a 10-day field experiment in which over 500 elite software developers were tasked with the challenge of developing a creative solution to a computational algorithmic problem from NASA. Participants were divided into two groups with identical skills distribution and exposed to the same competitive institutional setting. One group, however had self-selected and sorted itself into the competitive institution (as opposed to an outside option) and the other group was assigned to it. We find that the sorting effect, controlling for skills and formal incentives, nearly doubles the performance of the solutions developed as compared to the unsorted group. The sorting effect motivated more effort for individuals across the skills distribution and was of the same magnitude as the provision of formal incentives. Contrary to expectations of intrinsic crowding-out theory, high-powered incentives, independently motivated higher performance, especially at the higher skill levels.

Keywords: Sorting, Incentives, Intrinsic Motivation, Tournaments, Contests, Creative Workers, Experiment

April 2011

Acknowledgements: We are grateful to members of NASA's Space Life Sciences Directorate, including Jeff Davis, Elizabeth Richard, Jennifer Fogarty and Bara Reyna, for assistance in identifying an appropriate computational engineering challenge. The TopCoder team, including Jack Hughes, Rob Hughes, Mike Lydon, Ira Heffan, Jessie Ford and Lars Backstrom provided invaluable assistance in carrying out all aspects of the experiment. This research particularly benefitted from thoughtful comments by Kenneth Arrow, Pierre Azoulay, Iain Cockburn, Peter Coles, Daniel Elfenbein, Silke Januszewski Forbes, Mara Lederman, Joshua Lerner, Gary Pisano, Al Roth, Sandra Slaughter, Scott Stern, Eric von Hippel, Heidi Williams, D.J. Wu, and Catherine Tucker. Shane Greenstein deserves special recognition, as this project and line of research was triggered by comments he made at the 2009 NBER Summer Institute concerning the allocation of different types of workers across different types of organizations in the economy. Eric Lonstein provided outstanding project management and research assistance. All errors remain our own. Kevin Boudreau would like to acknowledge the financial support of the LBS M-Lab and a Google Faculty Research Grant. Karim Lakhani would like to acknowledge the generous support of the HBS Division of Research and Faculty Development and a Google Faculty Research Grant.

1 Introduction

The standard economic approach to eliciting optimal effort from workers is to offer pay-for-performance incentive schemes that reward workers for their observable outputs (Haris and Raviv 1978; Hölmstrom 1979). However, when it comes to creative workers engaged in novel problem solving and innovation-related tasks, high-powered incentives have been generally thought to be problematic. Scholars have noted that the encouragement of the risky uncertain search for solutions and repeated trial-and-error experimentation – essential for effectiveness in creative work – may be harmfully dampened if payments are directly contingent on every outcome (Hölmstrom 1989; Manso 2011; Azoulay, Graff Zivin, and Manso 2009). It is also very difficult to find sensible observables on which contracts can be written given that the creative process is non-routine and subtle in nature, with long gestational periods, and uncertain outcomes (Hölmstrom 1989; Aghion and Tirole 1994). Further complicating the governance of creative workers is the important role of intrinsic motivations in driving innovative performance. The application of high-powered, pay-for-performance incentives have been theorized and found to crowd-out and reduce the performance of intrinsically motivated creative workers (Lepper and Greene 1978; Frey 1994; Kreps 1997; Boneabeau and Tirole 2003; Ariely, Gneezy, Lowenstein, and Mazar 2009).

An added challenge of designing incentive schemes is that beyond governing the conduct of workers, the particular institutional details of incentive schemes also enable and promote worker self-selection and thus serve to attract particular sorts of workers, affecting the composition of the workforce (Salop and Salop 1976, Rosen 1986, Lazear 2000, Besley and Ghatak 2004, Dohmen and Falk 2011). A stable finding from both field and experimental data is that the nature of the incentive regime generates sorting on the basis of skills or productivity of workers; and in particular, higher skilled workers prefer contingent incentive programs (e.g.: Lazear 2000; Dohmen & Falk 2011).

However, emerging evidence—particularly from creative industries—suggests that sorting of workers is also on the basis of one's intrinsic preferences to work in a given institution or regime, i.e. their intrinsic "fit" with available institutions. For

example, academic science attracts (and socializes) individuals who deeply value autonomy and connections to a broader community (Dasgupta and David 1994; Aghion Dewtripoint, and Stein 2008); who are even willing to sacrifice monetary remuneration to get access to these institutional details in the private sector (Stern 2004). Indeed it may be that any field that requires individuals to engage in creative problem solving, may foster workers to exhibit very specific and distinctive preferences for particular institutional details. Studies in a range of contexts – such as architecture (Brain 1991, Nasar 1999); arts (including classical music, drama, film, fashion, music, etc.) (Throsby 1994, Caves 2000); law (Weisbrod 1983); non-profit and public sector (Besley and Ghatak 2004); and software development (Beecham, Baddoo, Hall and Robinson 2008) — each point to workers being greatly attuned to the institutional details of their work environment and the nature of the organizations they work for. As a consequence, different types of creative workers self-select and sort into diverse sorts of organizations.

In this article, we present results of a field experiment that we organized to elucidate and quantify how this intrinsic sorting and fit of creative workers with the regime in which they work impacts effort exerted and problem-solving performance — apart from the effects of the application of incentives or variation in worker skills. To emphasize, our interest here is to measure effects of the fit between a worker's intrinsic preferences and the regime or institutional "rules of the game" *per se*.

We designed and executed a 10-day field experiment in which our subjects, over 500 elite software developers, created solutions for a complex algorithmic engineering problem (from NASA's Space Life Sciences Directorate) while working in a competitive contest regime (top 5 competitors in groups of 20 could collect prizes). Our experimental context enabled us to obtain skill, effort and objective problem solving performance measures. We devised a sorting experimental approach that allowed us to create two groups of these elite software developers with each group having the identical distributions of raw problem-solving skills. However, one ("sorted") group uniformly preferred to work in the competitive contest regime instead of an outside option regime that involved working in a team. The other ("unsorted") group simply held the population-average preferences. To calibrate the size of effects in relation to the application of extrinsic monetary incentives, in some groups of 20 we allowed

participants to compete for $1000 in prizes; in other groups they competed for $0. The overall prize pool at stake was $25,000.

The thrust of our analysis involves comparing the activity and effort exerted by the subjects in the sorted and unsorted groups and the resulting problem-solving performance. Our main finding is that sorting on the basis of institutional preference nearly doubled our precise measure of problem solving performance, as compared to the unsorted control group, while holding constant skills and extrinsic incentives. The magnitude of this effect was statistically indistinguishable from the effect of competing for a $1000 cash prize, in a group of 20 competitors, as opposed to none. We found no interactions between sorting and cash incentive effects in any of the regressions presented in our analysis. Workers, sorted on the basis of their institutional preference for competition, choosing to work more hours on the problem explained this performance difference. The effect of cash incentives, however, worked rather differently than the sorting effect: the effects of cash incentives were most acute for highest-skilled workers and less so for "lower-skilled" workers (lower-skilled drawn from a pool of all relatively high-skilled software programmers); the sorting effect across lower- and higher- skilled workers was constant. The cash incentive effect acted most acutely through encouraging the proportion of subjects worked more than the minimum amount of time (again, with greater impact on the highest-skilled workers). The sorting effect worked both by increasing the proportion working more than the minimum and by increasing performance and effort levels of individuals, conditional on their having chosen to work more than the minimum level.

Therefore, we show that, the institutional arrangement in this context served to both independently motivate and to sort workers—and sorting not only systematically affected skills composition, but also affected the "fit" of workers and their behaviors and choices and effort, once having joined. Methodologically we implement a novel field experiment design that untangles the role of institutional fit, holding skills constant, and while varying cash incentives. We hence extend the sorting experiments approach pioneered in a series of lab experiments (*e.g.*, Bohnet and Kübler 2005; Camerer and Lovallo 1999; Lazear, Malmendier and Weber 2011; Cadsby, Song and Tappon 2007; Dohmen and Falk

2011). We also bring this tradition to a field setting, where subjects worked on a real, cognitively demanding problem, which required a creative solution.

The remainder of our paper is organized as follows. The next section reviews the relevant literature on sorting and self-selection. Section 3 discusses in detail the sorting field experiment approach used in our study. Section 4 describes our data and Section 5 contains the main results and various robustness tests of our findings. We conclude in Section 6.

2 Related Literature on Self-Selection and Sorting Effects

Prior work on institutions and incentive regimes has begun to tease out the role of self-selection and sorting of workers. Salop and Salop (1976) identified the importance of worker self-selection into incentive schemes that rewarded fast or slow turnover. Jovanovic (1979) showed that worker turnover in the economy is driven by individuals trying to find a match and fit between their own productivity and the nature and type of work in a firm. Workers try to find and stay in jobs where they are going to be relatively highly productive and self-select out of situations where they have low productivity. Rosen (1986) developed a theory of equalizing difference by emphasizing that the different tastes and preferences of workers results in diversity of employment choices and wages. Hence, Weisbrod (1983) has argued that the large (up to -40%) wage differential between lawyers specializing in public interest litigation compared to other types of traditional law practices can be accounted by an individual taste for public service and notoriety while controlling for differences in age, law school quality and academic performance. Besley and Ghatak (2004), in the context of mission-oriented organizations (non-profits and public administration), reasoned that a matching between mission preferences of agents and organization results in efficiency and an economizing on the need for high-powered incentives.

Empirical work on the importance of institutional fit on worker performance has attempted to differentiate the "treatment" effect of inducing new behaviors of given workers by changing the rules of the game from the self-selection and sorting effect. The main message from a range of studies, involving manual labor, is that self-sorting into a variable-pay incentive scheme is driven by higher worker skill resulting in improved

productivity. For example, Lazear (2000), using field data from a car windshield installation firm, showed that for manual labor, changes in the incentive scheme from hourly wages to piece rate improved firm productivity by 44%. He found that half of the productivity improvement was could be attributed to the new incentive system and the other half by changes in the workforce – where higher skilled workers were attracted to work at the firm and enjoy the benefits of piece rate. Further natural and field experiments in the context of tree planting and garment workers have provided additional support for this skill-based sorting effect (Shearer 2004; Franceschelli, Galiani and Gulmez 2009; Shi 2010).

Closely related to our work is Dohmen and Falk's (2011) study showing sorting effects in a laboratory setting involving subjects multiplying single and double-digit numbers. They found that higher skill workers prefer contingent-incentive schemes like tournaments or piece-rates to fixed payments, which also drove higher levels of outputs and effort. In a similar vein, laboratory experiments by Cadsby, Song and Tappon (2005) and Ericksson, Teyssier and Villeval (2005) demonstrate that higher productivity workers prefer to work under contingent-payment schemes and this also results in improved performance outcomes.

Laboratory experiments in behavioral economics have further shown that individuals have differential tastes for institutional regimes implying the significance of sorting and self-selection for a variety of related outcomes. The salience of sorting has been studied in prisoner's dilemma games (Bohnet and Kübler 2005), bargaining games (Oberholzer-Gee and Eichenberger 2008; Lazear Malmendier and Weber 2011), the gift-exchange game (Eriksson and Villeval 2008) and market-entry games (Camerer and Lovallo 1999).

3 Design of the Sorting Experiment

In the remainder of the paper, we present the design and results of a field experiment to estimate how workers' preferences to work within a given sort of institutional context influences problem-solving effort and outcomes. The essential idea is to compare a "sorted" group, one whose members uniformly prefer to work within a given regime, to an "unsorted" group who simply possess the population average distribution of preferences. Thus, rather than attempting to expose identical groups to

different treatments, as is usual experimental approach, this sorting experiment does just the opposite: it exposes groups of groups who systematically differ in a particular way (while held identical in their skills distribution) to identical treatments. The main activity consisted of sorted and unsorted participants competing to solve a computational-engineering problem over the course of ten days.

3.1 Field Setting

The inherent objective in a sorting experiment as ours is to demonstrate the importance of different types of participants on outcomes, and particularly the interaction between institutional preferences and institutional environment. It follows that the types of participants in question and the institutional context have some empirical relevance. For this reason, we pursued a field setting rather than a laboratory one. At the same time, the estimation of sorting effects here places especially high requirements of observing relevant microeconomic variables and doing so within a controlled environment.

We conducted the experiment on TopCoder.com, an on-line platform on which elite programmers from around the world sign-up as members and then compete against each other in a regular stream of contests to solve software development and computational-algorithmic problems for a variety of firms. Winners receive cash prizes, typically on the order of several hundred dollars. (See Boudreau, Lacetera and Lakhani (2011) for an extensive description of the context.) TopCoder insists on maintaining high fidelity records on all contests and participants. Thus when members directly compete against one another, TopCoder selects winners through an objective computationally-based scoring criterion with no performance ambiguity and all the results of the contests are publicly displayed. Furthermore, after each contest, each participant is given a precise and public ranking and skill rating for that particular problem type. The TopCoder rating is based on the long-established "Elo" system used to evaluate, rate, and rank chess grandmasters (Van Der Mass and Wagenmakers 2005) and other competitive contexts like the US College Bowl systems, National Scrabble Association members and the European Go Federation. The Elo rating creates relative performance metric based on the performance of all other participants and an individual's current and past performance. Thus at any given point in time, a participant has a clear idea of their ranking and rating within the entire population of TopCoder participants (See Appendix 1 for a view of the

public profile and ratings for a competitor). TopCoder adds further differentiation to the rating system by color-coding rating ranges to allow for easier identification and sense of achievement for the participants (red is the highest rated band). Interviews with TopCoder executives and competitors indicate that the TopCoder skills rating is often used as a credible signal in personnel hiring decisions by information technology (IT) intensive firms. Organizations like Google, Facebook and the US National Security Agency often encourage job applicants to obtain a TopCoder rating in order for them to be considered seriously for an open position.

In the experimental set-up, we sought to follow the routine characteristics of the usual TopCoder contest as much as possible. Just as TopCoder often does, we divided participants into 20-person groups who would compete directly with one another to solve a real problem in virtual competition "rooms." Participants use the TopCoder "arena" interface to program their solutions and to also observe the competitive field in their rooms. Information on direct competitors is updated in a side window of the interface, while the problem-solving screen is in the center of the screen. (See Appendix 2 for screen shots.) The side window lists the 19 other competitors' TopCoder unique "handles" (pseudonyms), and their numerical skill ratings that TopCoder routinely updates on the basis of past performance in competitions on the platform. The side window also displays the best scores for submitted solutions to that point. Clicking on any name reveals a complete history of the participant on the platform (Appendix 1).

Over the course of contest, individual competitors could submit as many times as they like. Each time a solution is submitted to the system, it is near-instantaneously subjected to a barrage of automated tests to register a score. Therefore, submissions provide a means of receiving feedback on interim solutions. Final scores for each participant were based on the highest score attained by the individual, where this was most always the last submission of the individual. Cash prizes were awarded on the basis of the rank order of scores attained in the room. First place in a group of direct competitors received $500, second place $200, third place $125, fourth place $100, fifth place $75. Therefore five of twenty competitors in each room received prizes.

Therefore, the TopCoder regime represents an institution with a distinctly competitive character, where individuals compete autonomously, have their performance

and skills objectively measured and shared publicly and awards are based explicitly on performance achieved and the subsequent ranking. Both the intensely competitive and autonomous characteristics of this context are quite salient to findings regarding the software developer labor market. Decades of descriptive and survey-based research has consistently reported considerable heterogeneity in psychological and behavioral orientations of software workers (Beecham et al 2008), with a large subset especially preferring to be autonomous "loners" (Schneiderman 1980) who crave individual rewards and recognition (Weinberg 1971, Couger and Zawacki 1980).

3.2 The Problem to be Solved by All Participants

The problem to be solved by each participant was to optimize the contents of the "Space Flight Medical Kit" for NASA's Integrated Medical Model (IMM). This is a computational-engineering problem to develop a robust algorithm that determines what components (consumable (e.g.: medicines) and non-consumable resources (e.g.: heart defibrillator) to include in the space medical kit included in each of NASA's space missions. There do exist algorithms that have been developed by NASA's internal engineers and life scientists; NASA's goal in participating in this experiment was to increase the sophistication and effectiveness over a wider range of applications where mission length would increase greatly (including missions to the International Space Station). (The winning algorithm from this experiment is now in use in all NASA missions.) The solution had to take into account that mass and volume are restricted in space flight, and that the resources in the kit needed to be sufficient to accommodate both expected and unexpected medical contingencies encountered while in space, lest the mission have to be aborted and an afflicted astronaut needs to be returned to earth. The content of the kit also had to be attuned to the characteristics of the space flight and crew and nature of the mission. The challenge was thus to develop an algorithm that addressed mission characteristics which traded off mass and volume against sufficient resources to minimize the likelihood of medical evacuation. (See Appendix 3 for a full problem statement and the scoring criteria.)

NASA also worked with TopCoder to develop a precise scoring function that would provide an objective performance metric for the code submissions from our subjects. The automatic scoring was based on an already established simulated set of

200,000 mission scenarios that had various medical contingencies that may occur during space flight. Each submission in the experiment was then subjected to a random set of 10,000 scenarios upon which performance of the algorithm and score were determined. The problem, being relatively focused, was to be solved as an integral project capable of being divided into a set of subroutines and call programs. The solution of this problem is not a matter of "software development" as we might casually think of, but is a non-trivial sort of algorithmic problem that participants in TopCoder tournaments frequently encounter.

3.3 Eliciting Preferences, Sorting and Matching Procedures

The central point of the design of this sorting experiment is to compare a "sorted" group of participants (one in which participants uniformly have a preference to work within the competitive TopCoder regime) to an "unsorted" group (one in which participants possess the population-average distribution of preferences). A key challenge here, however, is that individuals' institutional preferences may be correlated with raw problem-solving skills. Indeed, past studies have found evidence that higher-skilled workers tend to have a greater likelihood of preferring competitive environments and high-powered incentive schemes (e.g.: Lazear 2000, Dohmen and Falk 2011). In our analysis, however, we are interested in how individuals' preferences per se influence outcomes, not how preferences might be correlated with skill levels. One way to account for skills when drawing comparisons between sorted and unsorted groups is simply to exploit TopCoder's skill rating measures, applying these measures as controls when making econometric comparisons. Our experimental design is intended to go further to deal with observable as well possibly unobservable characteristics with an assignment procedure that involves a combination of matching and randomization. Figure 1 summarizes our assignment procedure.

<Insert Figure 1 Illustration of the Assignment Procedure>

As a first step to assuring that sorted and unsorted groups will have identical skills distributions, we begin by rank ordering all participants according to their TopCoder skill rating. With this rank-ordered list, we create successive "ordered pairs", from top to

bottom, or sets of two consecutive participants in terms of skill level. We then split the overall population into two equally sized groups of participants with identical skills distributions, simply by randomly assigning members of each ordered pair to one group or the other (i.e., group "A" and group "B" in Figure 1). To construct the sorted group we secretly asked members of just one of these groups about their preferences for working in the TopCoder regime. We followed past experimental work involving sorting (Dohmen and Falk 2011; Ericksson, Teyssier and Villeval 2005; Lazear Malmendier and Weber 2011**)** by presenting alternative choices and asking half of our participants to choose. However we diverged from past work by attempting to elicit our subjects' preferences without implying that a statement of preference would necessarily lead to an assignment of their choice. This was accomplished via asking participants to state their preference for a regime on a likert scale under three different hypothetical scenarios. The ordering of the likert scale choices was randomly reversed to prevent any sort of order preference and recency bias. (See Appendix 4 for the instrument used). Our aim was to minimize any altered behavior that might result from the solicitation of preferences. (See Section 5.2 on Hawthorne effects for more discussion on this point).

To elicit participant preferences as regards the competitive TopCoder regime, we presented members of group A with an alternative concept of competing on a "team" or "cooperative" regime, as the outside option. In the outside option, rather than competing among 20 individuals, participants would join a team of five individuals who would compete against four other teams. Total cash prizes would remain the same as in the usual competitive regime, but would be divided among team members. See Table 1 for details contrasting the TopCoder competitive regime and the cooperative outside option. To construct an unsorted group with the same skills distribution as the sorted group, as shown in Figure 2, but with institutional preferences that reflect the population average distribution of preferences, we simply assigned the ordered pairs (in group B) of those who expressed a preference for the competitive TopCoder (in group A). We did this by assigning these participants to the regime without regard to the institutional preferences of members of group B, or even without asking these preferences. (Participants who preferred the outside option and their ordered pairs therefore drop out of the sample.)

<Insert Figure 2 Kernel Density Skills Distribution, for Sorted and Unsorted Groups>

After constructing these larger pools of sorted and unsorted participants from groups A and B, the sorted groups were then randomly assigned to virtual "rooms" of 20 participants who would be direct competitors. Among these rooms of 20 sorted competitors, it was then randomly determined which would compete for $1000 in cash prizes (rather than $0). To construct rooms of unsorted competitors to which these would be compared, the ordered pairs of these sorted participants were then assigned to "mirror" rooms, to allow us to examine these ordered pairs under conditions of identical prizes and identical distribution of skills of competitors.

4 Data Set

Following the assignment procedure described in Section 3, the sample includes 516 observations (individual participants). There were originally 1040 individuals who participated in the overall event. Half of these individuals (520) were asked their preferences for the TopCoder competitive regime versus the cooperative outside option. Of these, 264 (50.8%) stated they preferred the competitive TopCoder regime over the cooperative outside option. These 264 participants were randomly assigned to fill up 13 virtual "rooms" (independent groups) of 20 individuals. Of the 13 rooms, 6 competed for a cash prizes of $1000 and the remainder did not. The ordered pairs of these assignees (who were not asked their preferences for the different regimes) were assigned to 13 rooms that mirrored these first 13, again with 6 rooms that competed for a cash prize. The number of observations (*i.e.*, individuals)--516--is not a perfect multiple of 20 (participants per room), as we dropped observations for which the algorithm skill rating was not available.[3]

As anticipated in Section 3, the fraction preferring the competitive TopCoder among those 520 individuals who were asked their preferences was positively correlated with skill level. Figure 3 presents a flexible non-parametric regression to illustrate the

[3] The equal treatment of individuals without algorithm ratings in the experiment was a requirement set forth by TopCoder.

proportion preferring the competitive regime at different levels of TopCoder's skill rating.

<Insert Figure 3 Proportion Preferring Competitive TopCoder Regime Over the Outside Option (Cooperative Regime), by Skill Level>

As regards to our research objective of measuring the effects of sorting, it should also be noted that the sample is itself drawn from the pool of TopCoder members. Therefore, the subsequent analysis of sorted and unsorted groups should be interpreted as somehow analogous to being "treating on the treated". Thus, we might speculate that any sorting effects we observe here could be small in relation to differences among more diverse groups. Our main dependent variable relates to problem-solving performance. A measure of the quality of each algorithm/solution was calculated with an automated test suite which assessed the performance of the submitted algorithm against a barrage of tests and contingencies, as was described in Section 3. The final score assigned to an individual competitor (*ProblemSolvingScore*) was the best of all submissions of a given participant, typically the final submission. Overall, 38% of the sample participants (195) made submissions. Non-submission received zero points. This led to a bimodal distribution in the sense that this 38% was relatively uniformly distributed up to a maximum score of 8957, with another 62% of observations spike at a score of zero. This sort of bimodality is also reflected in measures of effort and activity, described below.

Apart from problem-solving performance, we collected measures of the effort and actions of participants. The measure *NumSubmissions* is an observational measure related to level of activity. It provides a count of the total number of submissions by a participant made over the course of the 10-day experiment. Submitting code in this fashion was virtually costless and resulted in instantaneous feedback. This is a direct indication of the intensity of development effort, all else being equal, as code submission reflected code testing and evaluation. We also collected a more directly-interpretable measure of effort; the number of hours worked over the course of the 10-day by each participant. The variable *HoursWorked* was a self-reported estimate of the precise number of hours worked over the course of the ten days. This was collected by means of a mandatory survey that was completed electronically, immediately following the experiment. The

survey was mandatory in the sense that it needed to be completed prior to learning final results, rankings and winners. Further, receipt of a commemorative t-shirt (including the individual's name on the roster of participants) was conditional on having completed the survey. Where we did not immediately receive a response, we followed up with personalized emails and phone calls to get near complete coverage. The data on hours worked suggest there to be a broad distinction between those who devoted less than one hour to this exercise and a continuum of hours worked, if greater than this amount. As an indication of the close relationship between the observational measure and survey based measure, the proportion of observations with non-zero levels are almost identical, 38% versus 39%.

Observations are also coded in terms of whether they correspond to the sorted group with an indicator variable, *SortedonPreference*, and a $1000 cash prize (rather than no cash prize whatsoever), *CashPrize*. Our measure of raw problem-solving ability, *SkillRating*, is based on TopCoder’s rating system. We use specifically the rating calculated for what TopCoder terms “Algorithm” matches, software solutions to abstract and challenging problems akin to the problem in the experiment. Tables 2 and 3 provide variable definitions and summary statistics

<Insert Table 2 Variable Definitions>

<Insert Table 3 Summary Statistics>

5 Results

5.1 Comparison of Simple Means

Given the design of the experiment, a comparison of mean outcomes should, in principle, provide meaningful comparisons. Therefore, we begin by simply comparing *ProblemSolvingScore* across different groups. The mean *ProblemSolvingScore* attained across participants during the 10-day experiment was a score of 1736, with considerable variation (standard deviation = 2802). The most important result of this article can be noted by comparing the mean score of sorted and unsorted groups: the average problem-

solving performance of sorted groups is almost twice as high (an increase of 83%) as the unsorted with equal skills, an average score of 2244 versus 1228. Table 4 provides further details by breaking-down outcomes by both sorted and unsorted groups and those that competed for $1000 of prizes or none.

Several additional patterns are immediately apparent. First, the large effect of sorting on institutional preferences exists both with and without the cash prize (1682 – 758 = 924 point difference without the cash prize; 2976 – 2070 = 906 point difference with the cash prize). Similar sorting effects can be seen in the case of activity and effort measures, as well. Whether with or without cash prizes, participants made 1.8 (*i.e.*, 2.58 - .78 or 5.38 – 3.55) more submissions on average in the group that was sorted on institutional preferences. The cases of cash prize and no cash prize was slightly more substantively (although not statistically) different in the case of the number of hours worked: in the case of no cash prize, sorted participants worked 6.7 more hours (*i.e.*, 10.16 - 3.48), on average; in the case of cash prizes, sorted participants worked 10.7 hours more (*i.e.*, 21.42 - 10.70), on average.

<Insert Table 4 Comparison of Mean Outcomes, Stratified by Treatment>

5.2 Regression Analysis, Robustness and Interpretation

Although the earlier comparisons means should provide meaningful results, analyzing the data within a regression framework enables us to more explicitly assess key assumptions of the design, and to more deeply interpret patterns in the data. Baseline OLS regression results, with robust standard errors, are reported in Table 5.

If the assignment procedure was effective and left no systematic differences across treatments, the estimates should be unchanged when we include skill controls. (The specifications here are also reviewed as they provide a basis for later regression models.) For ease of comparison, model (5-1) begins by reporting the two-way correlation of *ProblemSolvingScore* on *SortedonPreference*. This effectively simply recasts the earlier descriptive statistics in a regression framework; the coefficient on *SortedonPreference*, 1016, is simply the difference between mean performance in sorted and unsorted groups (i.e., the difference between 2244 and 1228, as above). Model (5-2) re-estimates the coefficient on *SortedonPreference* with *SkillRating* now included as an explicit control

variable. The estimated coefficient is virtually unchanged. (The estimated constant term dramatically changes and becomes statistically indistinguishable from zero, given the importance of *SkillRating* in explaining performance outcomes.

To account for possible non-linearities in the relationship between skill and performance, we replace the linear control for skills with a series of dummies for different bands of skill levels. These bands correspond to different bands of skill levels (i.e., *SkillRating* in the following bands: <900, 900-1200, 1201-1500, 1501-2200, >2200) that TopCoder uses to distinguish different classes of competitors. As reported in model (5-3) the coefficient on *SortedonPreference* is again virtually unchanged. Model (5-4) provides a most stringent skill control by re-estimating the sorting effect by directly comparing the differences between the ordered pairs. (Recall that these pairs were based on matching individuals with effectively identical skills ratings, and then randomly assigning one of the sorted group and the other to the unsorted group.) This estimate of the effect of sorting based on "ordered pairs differences" is almost identical to the earlier estimates, again estimating a roughly 1000-point average effect of sorting.

<Table 5 Baseline OLS Regression Results>

Given the assignment procedure, the assignment to rooms with cash prizes should also be uncorrelated with skills or sorting. Indeed, including *CashPrize*, as in model (5-5) again leaves the coefficient on *SortedonPreference* statistically unchanged. (Note, the coefficient on *CashPrize* cannot be estimated with ordered pair differences, given ordered pairs were subjected to the same cash prize treatment. Therefore, the earlier-described dummies for different ranges of skill ratings were used instead.) At least as important, the inclusion of the *CashPrize* also provides another tangible indication of the relative importance of the sorting effect, this time in relation to the presence or absence of a formal high-powered incentive. While the point estimate of the coefficient on *CashPrize* (1324) is larger than that of *SortedonPreference* (1010), the difference is not statistically significant.

The comparison of simple mean outcomes across sorted and unsorted in Section 5.1 suggests a close link between levels of problem-solving activity (*NumSubmissions*,

HoursWorked) and performance (*ProblemSolvingScore*). To more explicitly reveal this link, we regress the two measures of activity on *SortedonPreference*. We again exploit the difference between matched pairs, this time by using a fixed effect for each matched pair in a robust count (Poisson) framework (Wooldridge 1999), as both measures of effort and activity are non-negative integers.[4] Results on are presented in Table 6. We begin with *NumSubmissions*, the observational measure of activity, and report results in model (6-1). The coefficient on *SortedonPreference* is estimated to be .65, implying an incidence rate ratio of 1.91. Model (6-2) reports an analogous model with our self-reported measure of effort and activity, *HoursWorked*, estimates a coefficient of .84. This implies an incidence rate ratio of 2.31. Clearly *NumSubmissions* has a natural advantage as a measure of effort and activity, given that it is an observational measure rather than self-reported. However, model (6-2) and the *HoursWorked* measure produced a better better-fitting model (log-likelihood of -2298 versus -707), and a more directly interpretable result. Further, it is possible that *NumSubmisssions* does not only capture effort and activity, but also captures say "style" of problem-solving (i.e., a heavy testing and iterative method, versus more contemplative and deliberate method). Therefore, *HoursWorked* is taken to be the preferred measure.

The large effect of sorting on the basis of institutional preference on effort and activity is perhaps analogous to large sorting effects found earlier on problem-solving performance. To further investigate the extent to which this boost in effort and activity can account for the boost in problem-solving performance, we again regress *ProblemSolvingScore* on *SortedonPreference*, but this time controlling for the level of effort and activity. (We also control for raw problem-solving skill, again using the most stringiest approach of estimating the effects on the basis of differences across matched pairs.) If effort and activity accounts for the performance boost, we should see the coefficient on the sorting variable to drop to zero, as we control for effort and activity. To roughly control for effort and activity, we simply include our preferred measure *HoursWorked*, along with a quadratic transform of this variable to allow for possible concavity or convexity. As reported in model (6-3), *ProblemSolvingScore* increases with

[4] Linear regressions produce similar results.
[6] Linear or quadratic terms of *SkillRating* are insignificant, if added to this model.

HoursWorked in an increasing and concave way, consistent with diminishing marginal returns to effort. Crucially, once effort levels are controlled---even in this simple way---the coefficient on *SortedonPreference* becomes statistically indistinguishable from zero. Therefore, evidence points to the sorting effect on problem-solving performance being mostly attributable to a boost in effort and activity.

<Insert Table 6 Results of Effort and Activity Regressions>

In Table 7, we report results in which we attempt to further interpret the precise nature of the sorting effect being measured. The experiment was designed with the intent of measuring the effect of sorting of individuals on the basis of individuals' preferences for types of institutions--the "rules of the game" *per se*. That is, individuals did not express preferences on the basis of *who* would be working within these regimes, as they simply did not know who would be assigned to their group. Nonetheless, it is still possible that, once assigned to a room, that the behavior of others in the room could have affected actions, incentives and activities of a participant (Bandeira, Barankay and Rasul 2005). For example, an especially active or challenging competitor might either stimulate or diminish the performance and activity of competitors in the same room. If so, then the earlier-measured sorting effects would not simply reflect a direct relationship between the individual worker and the rules of the game under which he functions, but also this social interaction. Therefore, we re-estimated the sorting effects, but this time controlling for the performance of the other 19 participants in the same room. We begin with the restatement of the results of model (5-3) for ease of comparison. Model (7-1) then adds a control for the average performance achieved, *ProblemSolvingScore*, for other participants in the same room. In the model, we control for the series of dummies for different skill bands. (Estimating on differences across matched pairs would eliminate most variation, as matched pairs were assigned to "mirror rooms" of pairs). Adding this additional variable to reflect possible social interactions finds nothing: the coefficient on this average is statistically zero and the coefficient on *SortedonPreference* is virtually unchanged. Model (7-2) adds a range of added measures of the distribution of performance of peers in the same room—variance, skew, maximum—and also finds no

change. We also ran analogous regressions, but with our measure of activity and effort, *HoursWorked*. Again, we found no change in results and no evidence of any sort of social interaction. We therefore conclude the estimated sorting effect does not include social interactions.

<Insert Table 7 Results of Tests for Social Interactions>

Finally, in interpreting the estimated sorting effects, it is important to assess whether the large effects measured here might have resulted from simply being asked their preferences, rather than necessarily subsequently assigning individuals according to their preferences. This would represent a Hawthorne effect of sorts. For example, if individuals believed that being asked their preferences for one regime or the other was tantamount to their being given the ability to choose their assignment, they might have then, say, had a sense of accountability or commitment to the choice (c.f. Dal Bo, Foster, and Putterman 2010) Our most important approach to mitigating this possibility was to design the process for eliciting preferences to avoid any direct implication that preferences would translate to assignments (Appendix 3). Individuals who were asked their preferences might also have had a heightened sense of being observed within an experiment. We attempted to diminish this effect by embedding the experiment within a "usual" TopCoder event, albeit one that took an especially high profile as a usual event (i.e., involving NASA, a large prize purse, ample publicity, etc.).

To explicitly estimate the magnitude of any Hawthorne effects, we attempted to compare outcomes of participants with similar preferences, but who were assigned to sorted and unsorted treatments. This was possible in a subset of out-of-sample data in which those who described themselves as "indifferent" were uniformly assigned to the cooperative team outside option. We found no statistical difference between either the *ProblemSolvingScore* or *HoursWorked* of indifferent participants in the sorted group and their ordered pairs who were in the unsorted group.

5.3 Synthesizing an Alternative Control Group with Propensity Scores

The estimated magnitude of the sorting effect will depend on the control group to which the sorted group is compared. The earlier regressions compared the sorted group

(in which 100% of participants preferred the regime) to an unsorted group whose preferences should be the same as the population distribution of preferences. Comparing a sorted group to this population average distribution is, of course, a natural and meaningful comparison to make. However, it is also true that high-skilled participants are more likely to prefer the competitive regime (Figure 3). Therefore our earlier estimates can be understood to underweight the effect of sorting among high-skilled participants given simply that the unsorted control group is more similar to sorted group among high-skilled competitors. In this section we generate an alternative "skills-neutral" estimate by synthesizing an alternative control group, in which the propensity to compete is fixed to the population average across all skills levels.

As a first step, we build a model of individuals' likelihood or propensity to prefer TopCoder's competitive regime over the outside option using data from the half of the original 1040 participants who were originally asked their preferences, prior to making assignments (cf. Figure 1). In a Logit model, we regress an indicator for a preference for the competitive TopCoder regime on a variety of demographic variables that were collected by TopCoder for all of its members (when they signed up to the platform). While raw problem-solving skill level is clearly an important predictor of institutional preference, it only explains a minority of variation; the intent here is to explain additional variation above and beyond this. (We will later reweight the control group according to these predicted institutional preferences.)

We present here a series of estimates, progressively adding the explanatory variables, with results presented in Table 8. The advantage of showing results with variables progressively added is to illustrate the stability of the model, despite widely varying specifications. While we do not need to interpret coefficients but only "fit" a model, the stability of the model lends support that model is somehow meaningful. We begin by simply regressing the preference for the TopCoder regime on skills, as in model (8-1). To allow for non-linearities, we specify *SkillRating* as the earlier-described series of dummies corresponding to the distinct ranges used by TopCoder.[6] Subsequent regressions add responses to a questionnaire TopCoder administers when new members sign-up to its platform. Model (8-2) introduces indicator variables corresponding to self-

[6] Linear or quadratic terms of *SkillRating* are insignificant, if added to this model.

reported reasons members initially joined the platform. Not surprisingly, those motivated by competition ("technology competition") reported systematically higher preference for the competitive TopCoder regime than for the cooperative outside option. Model (8-3) introduces indicator variables for different age ranges, finding older participants tend towards to the competitive regime. Model (8-4) introduces an indicator that distinguishes professionals from students and finds no statistically significant effect (although it remains consistently positive, when including or excluding other variables). Model (8-5) introduces a series of dummies to capture participants' countries of origin. Even in this quite radical re-specification of the model, where dummies for these 79 countries "soak up" much variation, the remaining model coefficients do not radically change—affirming the robustness of this probability or propensity model. In the analysis to follow, we assayed models (8-3), (8-4) and (8-5) as propensity models. We report results using model (8-4), given it is includes a large number of explanatory variables while remaining transparent in the nature of relationships that are exploited.

Reweighting to Establish a Constant Average Propensity Across all Skill Levels

Model (8-4) is then used to estimate the unobserved propensities of those in the unsorted group, who were not asked their preferences. We do so by substituting these participants' own demographic data into model (8-4). We then reweight the data to shift the mean propensity to competition to be equal to the aggregate population-average—across all skill levels.

<Insert Table 8 Logit Model Results of Probability / Propensity to Prefer Competitive Regime>

To reweight the control group in a way that adjusts propensity to compete while holding the skills distribution constant, we first divide the observations of the unsorted control group into ascending *SkillRating* blocks, following TopCoder's established color-coding of skill levels (i.e., <900, 900-1200, 1201-1500, 1501-2200, >2200). We then adjust the within-block relative weights of observations in order that the total within each block fixes the weighted average of propensities to the population average (i.e., the overall likelihood of preferring the competitive regime is 50.7%, from those who were

asked). To emphasize, the re-weighting occurs within blocks, while the total weight of each block is kept constant.

To describe this process, let each observation of estimated propensity to compete *P*, indexed by *j*. Within each skills block, indexed by *i*, there are N_i. We re-weight observations within each block in linear proportion to the propensity level, *i.e.*, $1+\omega_i \cdot P_{ij}$. Therefore the entire weighting scheme reduces to estimating the ω parameter for each skills block:

$$\omega_i = \frac{\overline{P}\cdot N_i - \sum_{j=1}^{N_k} P_{ij}}{\sum_{j=1}^{N_k} P_{ij}^2 - \overline{P}\cdot \sum_{j=1}^{N_k} P_{ij}}$$

The overall weight of each re-weighted block within the control group then also kept equal to its original overall weight, so as to leave the skills distribution unchanged.[7]

We proceed to re-estimate effects with this alternative (re-weighted) unsorted control group. To most explicitly reveal the effect of re-weighting, Figure 4 plots results of a flexible non-parametric regression of *ProblemSolvingScore* on *SkillRating* for both sorted and unsorted groups. For the unsorted group, we plot the relationship for both the un-weighted and re-weighted unsorted control group. As should be expected, the simple fact that the unsorted group is more similar to the sorted group at high skills leads there to be a seeming negligible performance difference between the sorted and unsorted group among high-skilled competitors. However, the re-weighted control group shows an roughly equal difference between sorted and unsorted groups across all skill levels.

<Insert Figure 4 Non-Parametric Regression of Problem-Solving Performance, Stratified by Treatment>

<Insert Table 9 Regression Results with Synthesized Control Group>

[7] To assure that the within-block reweighting did not systematically bias the skills distribution (by, say, systematically weighting observations on one "side" of each block), we confirmed that the un-weighted and re-weighted unsorted control group possessed statistically identical estimated means, variance and skew of skills. We also explicitly plotted the skills distribution (i.e., kernel density) of un-weighted and re-weighted data and found them to be almost identical and having no indication of any such systematic distortion. Of note, other approaches to re-weighting that would also hold skills constant while fixing the probability of preferring the competitive regime were possible; however, this approach will allow us to simultaneously reweight the skills distribution in a later analysis.

We then summarize and further explore these effects within a regression framework. Table 9 begins by re-stating the results of the un-weighted model (5-5). This model not only estimates the sorting effect, but includes *CashPrize* and therefore allows us to compare this effect. Further, this model controls for skills with the series of dummy variables corresponding to ranges. This is appropriate here, as it is no longer appropriate to estimate effects on the basis of differences across matched pairs once we re-weight the unsorted control group in subsequent steps. Model (9-1) re-estimates this model with the synthesized control group, which holds propensity to compete even across skill levels. The newly estimated coefficient on *SortedonPreference* is statistically unchanged, but is substantially increases by 13% (from 1010 to 1140). The estimated response to the cash prize changes far less leading the re-weighted estimates to be substantially closer. Given our interest in the overall distribution of effects, model (9-2) interacts the main explanatory variables with skill levels.[8] The model confirms earlier estimates of the magnitude of the sorting effect and no interaction effect (at least once the unsorted control group is reweighted, as can be seen in Figure 4). However, it appears the cash prize incentive operates quite differently, with higher-skilled participants responding far more to the cash prize than lower-skilled participants. This is consistent with higher-skilled individuals simply having a greater expectation of winning. The direct, un-interacted effect of the cash prize in this model is estimated to be statistically indistinguishable from zero.

5.4 Analysis of the Distribution (Bimodality) of Outcomes

To this point in the analysis, we have focused on estimating average effects of sorting on the basis of workers' institutional preferences (holding other factors constant). However, the earlier description of data (cf. Section 5.1) highlighted that outcomes were bimodal: a fraction of participants worked no more than a minimum amount of time (i.e., *HoursWorked* < 1 hour) and, consequently, received a zero score. Other participants worked more than this minimal amount and achieved a relatively smooth distribution of

[8] There is not significant effect in the remaining possible interaction, that between the cash prize and sorting, in all remaining analyses (not reported).

performance outcomes. Therefore, to better understand and describe this bimodality, the analysis here decomposes the effect of sorting on the decision to exert more than the minimum level of effort[9] from the effect on performance, conditional on having chosen to exert more than the minimum level of effort.

The Decision to Exert More than the Minimum Effort

Figure 5 begins by examining the decision to exert more than the minimum level of effort (*HoursWorked* > 1). Relationships are plotted separately for the sorted and unsorted groups. Again, we present the un-weighted and re-weighted unsorted control group. The flexible, non-parametric regression lines in this figure essentially trace the fraction of participants choosing to exert more than one hour of work. The patterns would appear to largely mirror the earlier observed patterns related to problem-solving performance (*i.e.*, Figure 4), with systematic differences between sorted and unsorted groups. The differences in overall performance across treatments would appear to at least then largely be due to the fraction of individuals simply choosing to exert some level of effort above the minimum.

Summarizing these differences in a regression framework allows us to essentially understand the overall (weighted) average effect, while comparing sorting and incentive effects. Linear probability models of choosing to exert effort are reported in first columns of . The unsorted control group in these regressions continues to be reweighted, as before. Model (10-1) regresses an indicator for exerting over one hour of effort on *SortedonPreference* and *CashPrize*, while controlling for the series of dummies for skill ranges. The estimated effect of sorting on institutional preference is a highly-significant 16%, on average. The estimated effect of providing a formal cash incentive in this same model was estimated to be substantially larger at 24%, but the difference between these two coefficients is not statistically significant. As before, we also include interaction terms with skills ratings, to provide deeper insight into the generation of the distribution of outcomes. Just as in the earlier case of overall performance, model (9-2), in model (10-2) we find no interaction between skills and sorting. And again, we find a strong positive

[9] If the analysis were simply descriptive, we might instead model the probability of achieving a problem-solving score greater than zero, and then the score, conditional on being greater than zero. However, the chosen approach better reflects the data generation process.

interaction between skills and the cash incentive (which, once included, erases the significance of cash prizes on their own).

<Insert Figure 5 Non-Parametric Regression of Probability of Working More than Minimum Level on Skills, Stratified by Treatment>

<Insert Table 10>

Problem-Solving Performance, Conditional on Exerting the Minimum Level of Effort

With the share of high-effort individuals (alternatively, the share who do not try in earnest) clearly an important contributor to the overall sorting effect, it remains to be determined whether the sorting effect appears in performance, conditional on having exerted more than the minimum level of effort. A simple comparison would suggest it does: the average *ProblemSolvingScore*, conditional on *HoursWorked* > 1, was 4596 in the sorted group versus just 4281 in the unsorted group, a difference of 315 points (7%). To provide more precise estimates, we study this comparison within a regression framework. (We focus here on problem-solving performance, *ProblemSolvingScore*, as the key dependent variable, conditional on exerting effort. However, the results are closely mirrored in our effort measures, given the close link among them.)

We follow the same basic re-weighting approach, as earlier (cf. Section 5.3). However, an important difference in this case is that here we are analyzing a subset of the sorted group and subset of the unsorted group. Therefore, we must recalculate propensity weights for the unsorted control group (subset), as distinct from the weights in the earlier analysis. The procedure is identical, however the data to which the procedure is applied differs. A second difference is these subsets are no longer identically distributed skills distributions. Therefore, while the relative weights of observations within each skills band of the unsorted control group (subset) are reweighted to fix the propensity to compete to the population average, the bands themselves are re-weighted to set the skills distribution of the unsorted control group (subset) to be the same as the sorted group (subset).

As above, we begin our analysis with a graphical presentation of differences between sorted and unsorted groups, as in Figure 6. (We do not present patterns conditional on skills in this case, as was done in Figure 4 or Figure 5, as the fewer data

points in this subset leads to far less precise estimates—particularly at high skills ratings, where there is already fewer data points in the full sample.) Perhaps the first and plainest pattern that can be appreciated from Figure 6 is that all distributions are relatively smooth and flat; not uniform, but "thickly" distributed across different problem-solving scores. Further, the sorted group is clearly distributed "to the right" of both the raw control group distribution and the re-weighted control group. The difference with the re-weighted control group is even greater because in the group of unsorted workers, it was the relatively high-skilled workers would chose to participate. Therefore, the re-weighting entails reducing the weight on these workers in the statistical comparison.

As reported in model (10-3), the estimated average effect sorting effect (once reweighting the control group, controlling for the presence of a cash prize and controlling for individual skills) is 1,301 points—representing about half a standard deviation in overall variation of *ProblemSolvingScore* (Table 3). The effect of the presence of a cash prize is estimated to be considerably smaller at 413 and the coefficient is statistically insignificant. In the case of problem-solving performance conditional on exerting above the minimum effort, we find no evidence of interactions, as in model (10-2).

6 Conclusion

In this paper, we report evidence from a 10-day "sorting" field experiment involving over 500 elite programmers engaged in trying to create a software solution to a real computational engineering problem from NASA. Our aim in this experiment was to investigate how self-selection and sorting on the basis of institutional preferences of workers impacted effort and performance for a creative problem-solving task, while accounting for both skills and extrinsic incentives. We find that the fit-based sorting between creative workers and their preferred institutional regime has significant economic effects – nearly doubling problem-solving performance. The underlying mechanism behind the drastically improved performance is the exertion of significantly more effort, across the skills distribution, when individuals get to fit. Sorting encouraged more workers to participate and to exert more effort on the problem-solving task. While the extant literature has identified that sorting on the basis of skill drives improved

productivity, we show that sorting on the basis of institutional preference is equally as important. Concurrent with Lazear (2000), we also report that in the case of creative workers attempting to solve a difficult problem, the sorting effect is of a similar magnitude as the provision of formal incentives in driving effort and performance.

Our experiment was uniquely able to separate out the impact of formal incentives on creative worker performance. We find that the presence of high-powered incentives was a strong motivating force driving effort and performance for our subjects. Indeed the effects of formal incentives were the strongest on the workers with high skill levels. The cash incentive operated by increasing the proportion of workers who exerted more than the minimum amount of time; with the greatest impact on the highest skilled workers. Our findings about the positive salience of high-powered incentives on the performance of workers departs from existing views about the potential crowding-out effect of intrinsically motivated creative workers. Contrary to the literature on the negative impact of pecuniary incentives on creative workers, we uncover that extrinsic rewards boost innovation performance in the case of elite software developers who are used to competing. Thus the negative incentive hypothesis in the creativity and innovation literatures may be limited to laboratory settings and to individuals not used to competing on a regular basis. Thus it may be possible that the crowding-out effect observed in the laboratory may be an artifact of randomization and could be countered through a selection experiment. Importantly this finding indicates that if creative workers are used to competition, for example architects, software developers, graphic artists, screenwriters, then variable incentive schemes can produce high performance instead of deterring effort.

Overall we hope that we have laid the groundwork for future investigations on the use of sorting experiments with creative workers in the field. The advent of internet-based platforms for real work tasks provides unprecedented opportunities for economists to test and extend theoretical insights by creating a controlled laboratory in the world.

References

Aghion, P, M. Dewatripoint, and J.C. Stein. 2008. "Academic freedom, private-sector focus, and the process of innovation." *RAND Journal of Economics* 39(3): 617-635.

Ariely, D., U. Gneezy, G. Lowenstein and N. Mazar. 2009. "Large Stakes and Big Mistakes." *Review of Economic Studies* 76(2): 451-469.

Azoulay, P, JG Zivin, and G Manso. 2009. "Incentives and Creativity: Evidence from the Academic Life Sciences." *NBER Working Paper #15466*.

Beecham, S., N. Baddoo, T. Hall, and H. Robinson. 2008. "Motivation in Software Engineering: A systematic literature review." *Information and Software Technology* 50: 860-878.

Benabou, Roland, and Jean Tirole. 2003. "Intrinsic and Extrinsic Motivation." *Review of Economic Studies* 70(3): 489-520.

Besley, T, and M Ghatak. 2004. "Competition and incentives with motivated agents." *American Economic Review* 95(3):616-636.

Bandiera, O., I. Barankay, and I. Rasul. 2005. "Social Preferences and the Response to Incentives: Evidence from Personnel Data." *Quarterly Journal of Economics*, 120(3): 917–962.

Bohnet, I., D. Kübler. 2005. Compensating the cooperators: is sorting in the prisoner's dilemma possible? *Journal of Economic Behavior and Organization* 56: 61–76.

Cadsby, B, C., F. Song, and F. Tapon. 2007. "Sorting and Incentive Effects of Pay-for-Performance: An Experimental Investigation." *The Academy of Management Journal*, 50(2): 387 – 405.

Camerer, C.F., and D. Lovallo. 1999. "Overconfidence and Excess Entry: An Experimental Approach." *American Economic Review*, 89(1): 306–318.

Caves, Richard E. 2002. *Creative Industries: Contracts between Art and Commerce.* Harvard University Press.

Couger, J.D. & Zawacki, R.A., 1980. *Motivating and managing computer personnel*, New York: Wiley.

Bo, P.D., A. Foster, and L. Putterman. 2010. "Institutions and Behavior: Experimental Evidence on the Effects of Democracy." *American Economic Review* forthcoming.

Dasgupta, P, and P A. David. 1994. "Towards a new economics of science." *Research Policy* 23: 487-524.

Eriksson, T. and MC. Villeval. 2008. "Performance-Pay, Sorting and Social Motivation." *Journal of Economic Behavior & Organization*, 68: 412–421.

Eriksson, T., S. Teyssier, and M. Villeval. 2009. "Does Self-Selection Improve the Efficiency of Tournaments?" *Economic Inquiry* 47(3): 530–548.

Franceschelli, I, S Galiani, and E Gulmez. 2010. "Performance pay and productivity of low-and high-ability workers." *Labour Economics* 17:317-322

Frey, BS. 1994. "How Intrinsic Motivation is Crowded Out and In." *Rationality and Society* 6:334-352

Gneezy, U., M. Niederle, and A. Rustichini. 2003. "Performance in Competitive Environments: Gender Differences." *Quarterly Journal of Economics*, 118(3): 1049–1074.

Harris, M, and Artur Raviv. 1978. "Some Results on Incentive Contracts with Applications to Education and Employment, Health Insurance, and Law Enforcement." *The American Economic Review* 68(1): 20-30.

Hölmstrom, B. 1989. "Agency costs and innovation." *Journal of Economic Behavior and Organization* 12(305-327).

———. 1979. "Moral hazard and observability." *The Bell Journal of Economics* 10(1):74-91

Jovanovic, B. 1979. "Job Matching and the Theory of Turnover." *The Journal of Political Economy* 87(5): 972-990.

Kreps, DM. 1997. "The Interaction Between Norms and Economic Incentives." *The American Economic Review* 87(2): 359-364.

Lazear, E.P. 2000. "Performance Pay and Productivity." *American Economic Review*, 90(5): 1346–1362.

Lazear, E.P., U. Malmendier, and R.A. Weber. 2011. "Sorting, Prices and Social Preferences." Unpublished manuscript.

Maas, Han L. J. van der, and Eric-Jan Wagenmakers. 2005. "A Psychometric Analysis of Chess Expertise." *The American Journal of Psychology* 118(1): 29-60.

Manso, Gustavo. 2011. "Motivating Innovation." *Journal of Finance* forthcoming.

Oberholzer-Gee, Felix, and Reiner Eichenberger. 2008. "Fairness in Extended Dictator Game Experiments." The B.E. Journal of Economic Analysis & Policy, 8(1). art. 16.

Rosen, S. 1986. "The theory of equalizing differences." *Handbook of labor economics*.

Salop, J., and S. Salop. 1976. "Self-Selection and Turnover in the Labor Market." Quarterly Journal of Economics, 90(4): 619–627.

Schneiderman, B., 1980. *Software Psychology: Human Factors in Computer and Information Systems*, Scott Foresman & Co.

Shi, L. 2010. "Incentive Effect of Piece-Rate Contracts: Evidence from Two Small Field Experiments." *The BE Journal of Economic Analysis & Policy* 10(1).

Stern, Scott. 2004. "Do scientists pay to be scientists?" *Management Science* 50(6): 835-854.

Throsby, D. 1994. "The production and consumption of the arts: a view of cultural economics." *Journal of Economic Literature*.

Weisbrod, B. A. 1983. "Nonprofit and proprietary sector behavior: Wage differentials among lawyers." *Journal of Labor Economics* 1(3): 246–263.

Figures

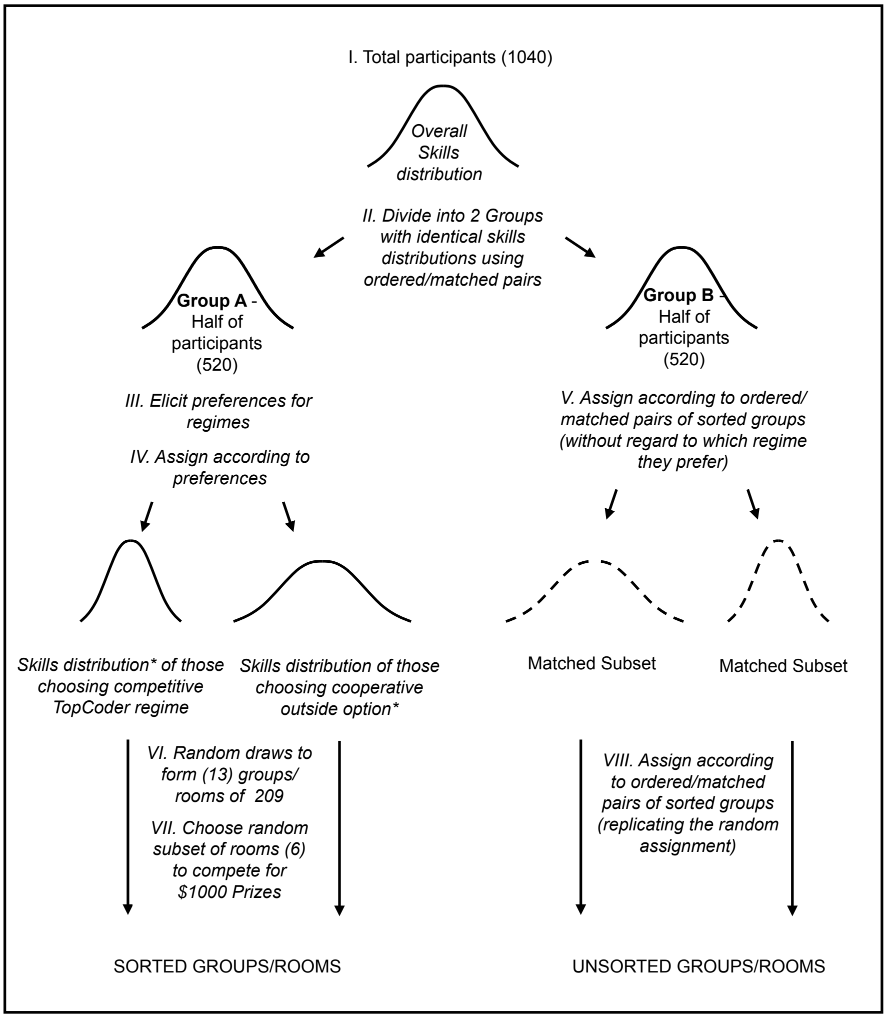


Figure 1 Illustration of the Assignment Procedure

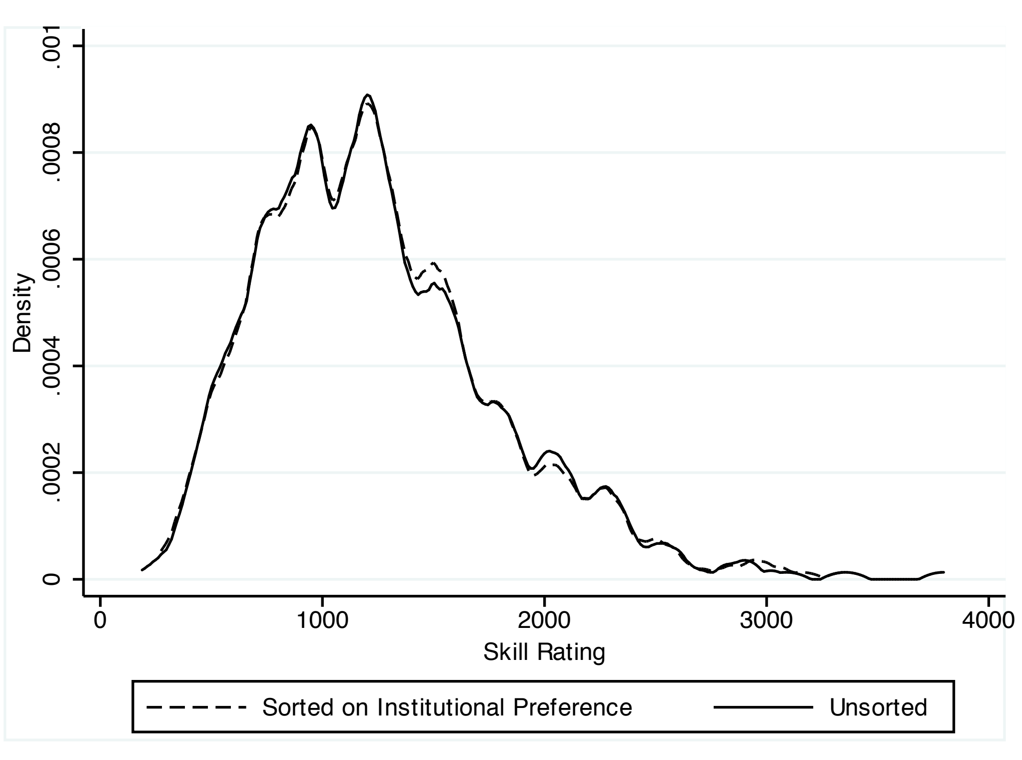


Figure 2 Kernel Density Skills Distribution, for Sorted and Unsorted Groups

Note: The lines in the figure are kernel density estimates of the frequency of observations across different levels of the problem-solving skill rating. The density is estimated with an Epanechnikov kernel. A very narrow bandwidth of 50 was chosen to highlight the closeness of the skills distributions.

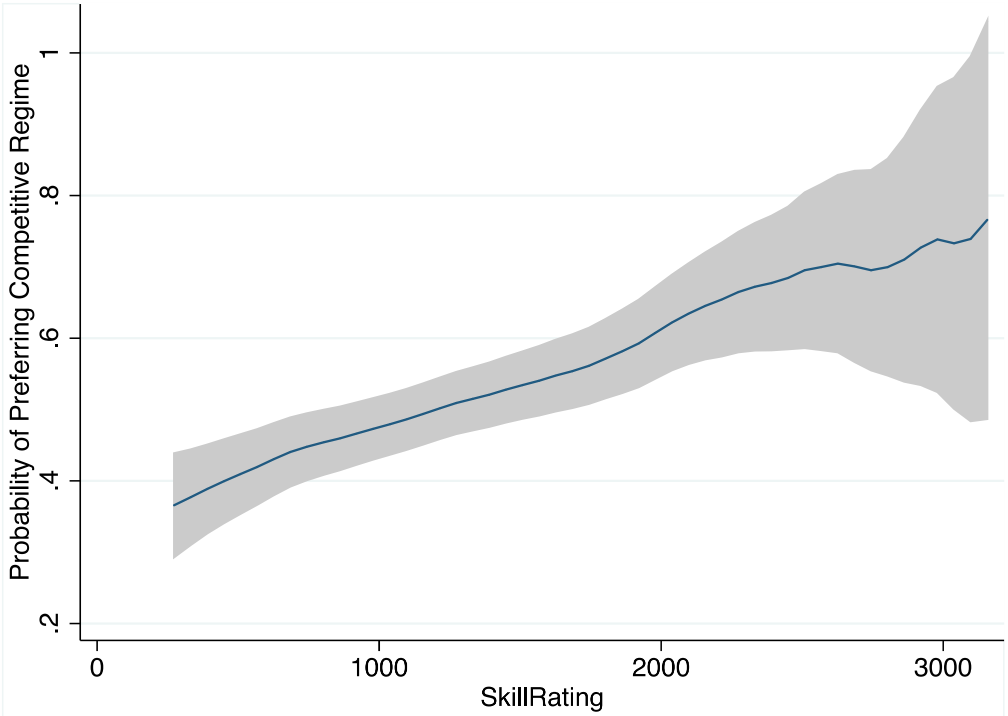


Figure 3 Proportion Preferring Competitive TopCoder Regime Over the Outside Option (Cooperative Regime), by Skill Level

Note: The line fits to a series of 1's and 0's depending on whether the individual preferred TopCoder's competitive regime (1) or the outside option (0). The line is a locally-weighted fitted second-order polynomial. The local weighting is based on an Epanechnikov kernel with a bandwidth of 300. The shaded grey region represents the 90% confidence interval for the estimate.

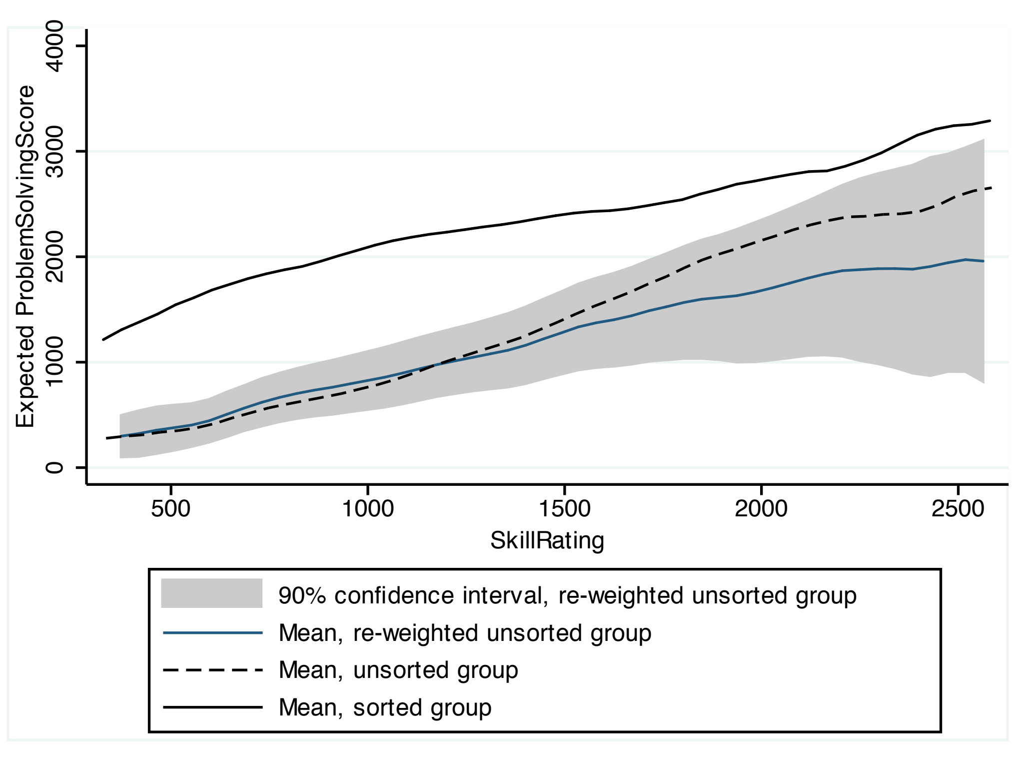


Figure 4 Non-Parametric Regression of Problem-Solving Performance, Stratified by Treatment

Note:Each of the lines fits a locally-weighted fitted second-order polynomial, with local weighting based on an Epanechnikov kernel with a bandwidth of 300. For each line, the relationship is fitted with a different sample. The solid black line is the relationship for the group that has been sorted on the basis of their preference for the TopCoder competitive regime. The dashed black line is the relationship for a group that has not been sorted on their preferences, but with an identical skills distribution. The blue line is the same unsorted group, but whose data points have been re-weighted according to steps described in Section 5.3 to maintain the population-average propensity towards the TopCoder competitive regime, while holding the skills distribution constant. The shaded grey region represents the 90% confidence interval for the estimate.

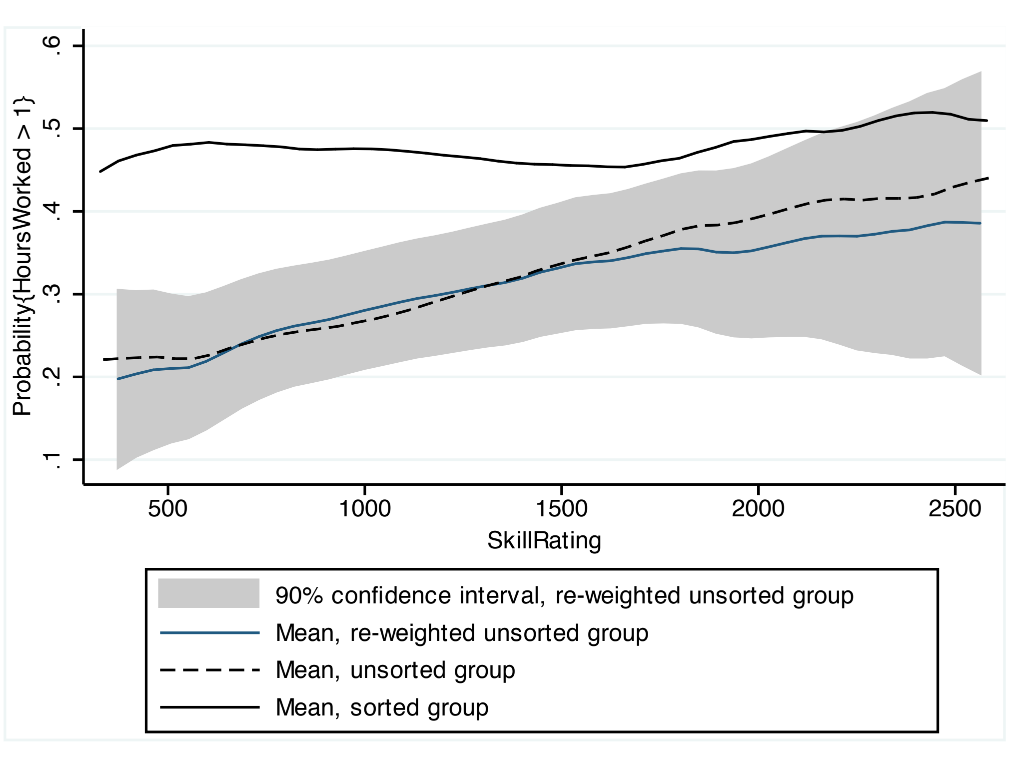


Figure 5 Non-Parametric Regression of Probability of Working More than Minimum Level on Skills, Stratified by Treatment

Note: Each of the lines fits a locally-weighted fitted second-order polynomial, with local weighting based on an Epanechnikov kernel with a bandwidth of 300. For each line, the relationship is fitted with a different sample. The solid black line is the relationship for the group that has been sorted on the basis of their preference for the TopCoder competitive regime. The dashed black line is the relationship for a group that has not been sorted on their preferences, but with an identical skills distribution. The blue line is the same unsorted group, but whose data points have been re-weighted according to steps described in Section 5.3 to maintain the population-average propensity towards the TopCoder competitive regime, while holding the skills distribution constant. The shaded grey region represents the 90% confidence interval for the estimate.

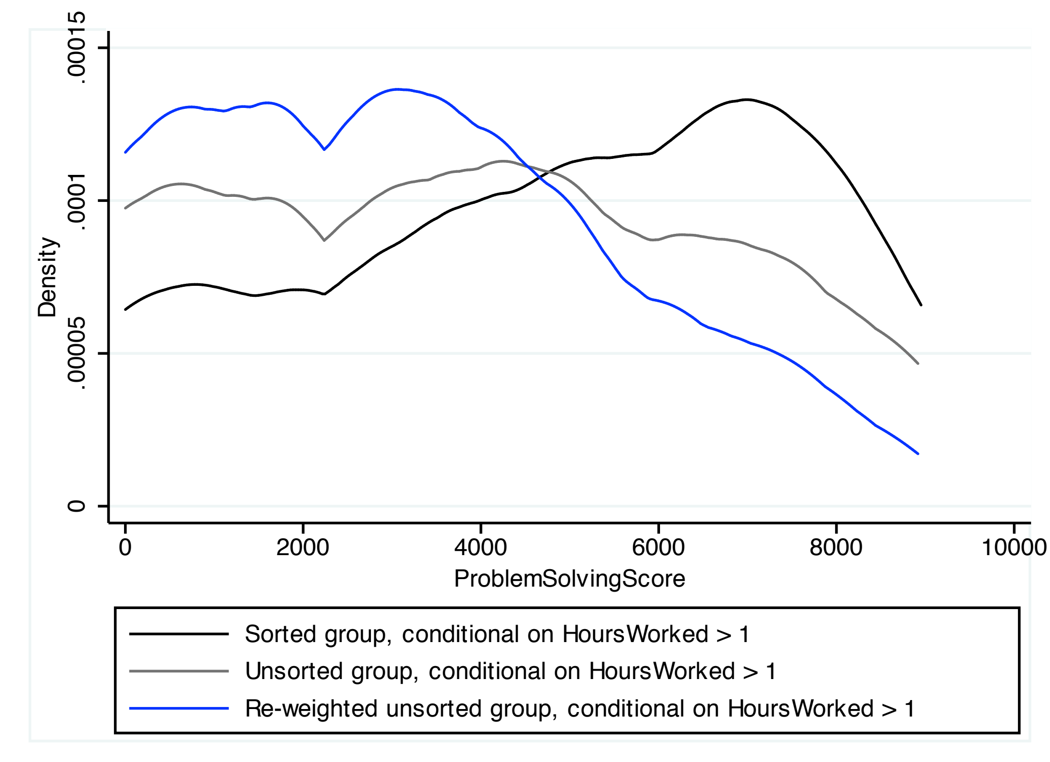


Figure 6 Kernel Density of Problem-Solving Performance, Stratified by Treatments

Note: The lines in the figure are kernel density estimates of the frequency of observations across different levels of the problem-solving skill rating. The density is estimated with an Epanechnikov kernel with a bandwidth of 1000. For each line, the relationship is fitted with a different sample. The black line is the density for the group that has been sorted on the basis of their preference for the TopCoder competitive regime. The grey line is the density for a group that has not been sorted on their preferences, but with an identical skills distribution. The blue line is the same unsorted group, but whose data points have been re-weighted according to steps described in Section 5.4 to maintain the population-average propensity towards the TopCoder competitive regime, while holding the skills distribution the same as that for the sorted group of the black line.

Tables

Table 1 Key Features of the Competitive TopCoder Regime and the (Cooperative) Outside Option Regime

	COMPETITIVE TOPCODER REGIME	OUTSIDE OPTION REGIME
Size of a Group	20 competitors	4 x 5-person teams (assigned)
Payoffs	Total: $1000, divided 5-ways	Total: $1000, divided 5-ways
	$500 – best submission	$1000 – divided among winning team members (according to average of team members' suggestions)
	$200 – second best submission	
	$125 – third best submission	
	$100 – fourth best submission	
	$75 – fifth best submission	
Communications & Code Sharing	None	A private team-message board and ability to send directed messages
Information	Competitors "see" who else is in the group, their ratings and top code submissions to date	Competitors "see" best scores to date of other teams; detailed information on the statistics and background of their own team members

Table 2 Variable Definitions

Variable	Definition
ProblemSolvingScore	Numerical score awarded to a solution as an assessment of overall quality, based on automated test suite
NumSubmissions	Number of solutions submitted to be compiled, tested and scored by an individual participant during the course of the experiment
HoursWorked	Number of hours worked by an individual participant during the course of the experiment
SortedonPreference	Indicator switched to one for participants who were asked their preferences regarding the regimes and subsequently assigned to their preferred regime
Prize	Indicator switched to one for participants within a group of 20 that competed for a $1000 cash prize
SkillRating	Measure of general problem solving ability in Algorithmic problems based on historical performance on TopCoder platform

Table 3 Means, Standard Deviations and Correlations

	Variable	Mean	Std. Dev.	Min	Max	(1)	(2)	(3)	(4)	(5)	(6)
(1)	*ProblemSolvingScore*	1736	2802	0	8957	1.00					
(2)	*NumSubmissions*	2.89	6.32	0	42	.74	1.00				
(3)	*HoursWorked*	10.80	20.68	0	190	.61	.55	1.00			
(4)	*SortedonPreference*	.50	.50	0	1	.18	.14	.21	1.00		
(5)	*Prize*	.43	.50	0	1	.25	.22	.22	.00	1.00	
(6)	*SkillRating*	1344	546	328	3354	.22	.17	-.02	.00	.02	1.00

Table 4 Comparison of Mean Outcomes, Stratified by Treatment

UNSORTED ON INSTITUTIONAL PREFERENCE					
NO CASH PRIZE			CASH PRIZE		
Variable	Average	Standard Deviation	Variable	Average	Standard Deviation
ProblemSolvingScore	578	582	*ProblemSolvingScore*	2070	3052
NumSubmissions	.78	2.50	*NumSubmissions*	3.55	6.76
HoursWorked	3.48	3.29	HoursWorked	10.70	17.17

SORTED ON INSTITUTIONAL PREFERENCE					
NO CASH PRIZE			CASH PRIZE		
Variable	Average	Standard Deviation	Variable	Average	Standard Deviation
ProblemSolvingScore	1682	2754	*ProblemSolvingScore*	2976	3214
NumSubmissions	2.58	5.92	*NumSubmissions*	5.38	8.53
HoursWorked	10.16	19.39	*HoursWorked*	21.42	30.32

Table 5 Baseline OLS Regression Results

	Dependent Variable = *ProblemSolvingScore*				
Model:	(5-1)	(5-2)	(5-3)	(5-4)	(5-5)
Explanatory Variables	Two-Way Correlation	Linear Sklls Control	Skills-Level Dummies	Matched Pair Differences	Prize Control
SortedonPreference	1,016*** (243)	1,016*** (237)	1,009*** (236)	1,042*** (235)	1,010*** (229)
Prize					1,324*** (239)
Skills-Level Dummies			Yes		Yes
Constant	1,223*** (153)	-248 (281)			
Adj R-Squared	.03	.08	.08	.12	.14

Notes. *, **, and *** indicate statistical significance at the 10%, 5% and 1% levels, respectively; heteroskedasticity robust standard errors reported; number of observations = 516 participants.

Table 6 Results of Effort and Activity Regressions

	Number of Submissions	*Hours Worked*	*Problem Solving Score*
Model:	(6-1)	(6-2)	(6-3)
Specification	Count Model, Multiplicative Matched Pair Fixed Effects		Linear model, matche pair differences
SortedonPreference	.65***	.84***	216
	(.19)	(.16)	(163)
HoursWorked			157***
			(16)
HoursWorked^2			-.85***
			(.20)
Log-Likelihood	-707	-2298	
Adj R-Squared			.55

Notes. *, **, and *** indicate statistical significance at the 10%, 5% and 1% levels, respectively; heteroskedasticity robust standard errors reported; number of observations = 516 participants.

Table 7 Results of Tests for Social Interactions

	Dependent Variable = ProblemSolvingScore		
Model:	(5-3)	(7-1)	(7-2)
Explanatory Variables	Matched Pair Differences		
SortedonPreference	1,009***	1,055***	1,082***
	(236)	(354)	(361)
Others in Same Room			
Mean		-.04	-.13
		(.24)	(.49)
Variance			.49
			(1.09)
Skew			-49
			(534)
Max			-.22
			(.31)
Skills-Level Dummies	Yes	Yes	Yes
Adj R-Squared	.08	.12	.11

Notes. *, **, and *** indicate statistical significance at the 10%, 5% and 1% levels, respectively; heteroskedasticity robust standard errors reported; number of observations = 516 participants.

Table 8 Logit Model Results of Probability / Propensity to Prefer Competitive Regime

Dependent Variable:	*Dependent Variable = I{Prefer Competitive TopCoder Regime*				
Model:	(8-1)	(8-2)	(8-3)	(8-4)	(8-5)
Explanatory Variables	Skills Level Dummies	Motivations	Age	Employed	Countries of Origin
<u>Raw Problem-Solving Skill</u>					
900 ≤ SkillRating < 1200	.54**	.52**	.52**	.53**	(.23)
	(.25)	(.26)	(.26)	(.26)	(.28)
1200 ≤ SkillRating < 1250	.47*	.45*	.45	.47*	.07
	(.26)	(.27)	(.28)	(.28)	(.29)
1500 ≤ SkillRating < 2200	.98***	.93***	.97***	.99***	.59**
	(.26)	(.27)	(.27)	(.27)	(.30)
2200 ≤ SkillRating	1.18***	1.15***	1.21***	1.24***	.81*
	(.43)	(.43)	(.43)	(.44)	(.47)
<u>Motivation for Joining TopCoder</u>					
"*Cash Prizes*"		.33	.26	.24	.20
		(.30)	(.31)	(.31)	(.32)
"*Employment Opportunity*"		-.28	-.33	-.41	-.44
		(.36)	(.36)	(.37)	(.39)
"*Technology Competition*"		.64***	0.53**	.50**	.45*
		(.22)	(.23)	(.23)	(.24)
<u>Age</u>					
18-28			.41	.37	-.35
			(.57)	(.57)	(.29)
25-34			.59	.37	-.55
			(.59)	(.63)	(.40)
35-44			1.32*	1.05	-.01
			(.70)	(.75)	(.61)
≥45			2.36*	2.18*	.98
			(1.24)	(1.25)	(1.16)
Declined to Answer			.57	.39	-.35
			(.84)	(.86)	(.76)
<u>Other</u>					
"*Professional*" *(versus* "*Student*"*)*				.28	.41
				(.27)	(.29)
Country of Origin Dummies					Yes
Log Likelihood	-345	-338	-333	-332	-319

Notes. *, **, and *** indicate statistical significance at the 10%, 5% and 1% levels, respectively; number of observations = 520 participants.

Table 9 Regression Results with Synthesized Control Group

	Dependent Variable = *ProblemSolvingScore*		
Model:	(5-5)	(9-1)	(9-2)
Explanatory Variables	Unweighted	Re-Weighted by Propensity for Competitive Regime	
SortedonPreference	1,010*** (229)	1,140*** (235)	1,198* (626)
Sorted x Skill			.03 (.45)
CashPrize	1,324*** (239)	1,360*** (246)	-1028 (643)
Prize x Skill			1.86*** (.50)
Skills-Level Dummies	Yes	Yes	Yes(*)
Adj R-Squared	.14	.07	.03

Notes. *, **, and *** indicate statistical significance at the 10%, 5% and 1% levels, respectively; heteroskedasticity robust standard errors reported; number of observations = 516 participants in the cooperative regime; 524 in the cooperative regime. (*) A linear control for SkillRating is included in addition to the dummies.

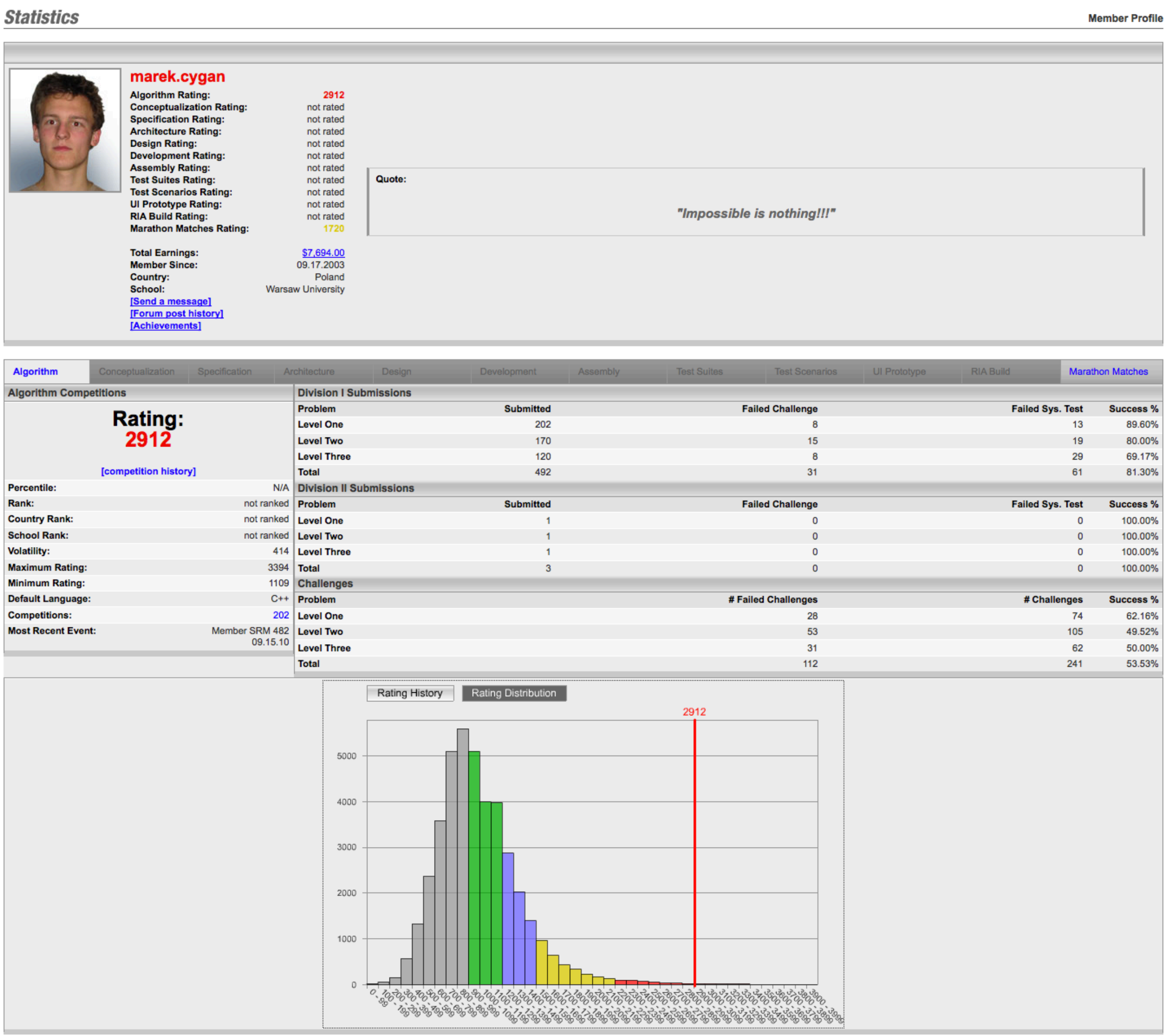
Table 10 Probability of Working Greater than the Minimum Amount and Problem-Solving Performance, Conditional on Working Greater than the Minimum Level of Effort

Dependent Variable:	*I{HoursWorked > 1}*		*ProblemSolvingScore | HoursWorked > 1*	
Model:	1	2	3	4
Explanatory Variables	Synthesized Control Group	Add Interactions with Skill	Synthesized Control Group	Add Interactions with Skill
SortedonPreference	.16*** (.04)	.31*** (.12)	1,301*** (416)	2,467* (1437)
Sorted x Skill		0 (0)		-.83 (1.05)
CashPrize	0.2424*** (.05)	-.12 (.12)	413.70 (391.07)	492.22 (1134.60)
Prize x Skill		0.0003*** (.00)		-.07 (.79)
Skills-Level Dummies	Yes	Yes(*)	Yes	Yes(*)
Adj R-Squared	.09	.11	.14	.14

Notes. *, **, and *** indicate statistical significance at the 10%, 5% and 1% levels, respectively; heteroskedasticity robust standard errors reported; number of observations = 516 participants in the cooperative regime; 524 in the cooperative regime. (*) A linear control for SkillRating is included in addition to the dummies.

Appendices

APPENDIX 1: Participant Ratings



Statistics — Member Profile

marek.cygan

Algorithm Rating:	2912
Conceptualization Rating:	not rated
Specification Rating:	not rated
Architecture Rating:	not rated
Design Rating:	not rated
Development Rating:	not rated
Assembly Rating:	not rated
Test Suites Rating:	not rated
Test Scenarios Rating:	not rated
UI Prototype Rating:	not rated
RIA Build Rating:	not rated
Marathon Matches Rating:	1720
Total Earnings:	$7,694.00
Member Since:	09.17.2003
Country:	Poland
School:	Warsaw University

[Send a message]
[Forum post history]
[Achievements]

Quote:

"Impossible is nothing!!!"

Algorithm | Conceptualization | Specification | Architecture | Design | Development | Assembly | Test Suites | Test Scenarios | UI Prototype | RIA Build | Marathon Matches

Algorithm Competitions

Rating:
2912

[competition history]

Percentile:	N/A
Rank:	not ranked
Country Rank:	not ranked
School Rank:	not ranked
Volatility:	414
Maximum Rating:	3394
Minimum Rating:	1109
Default Language:	C++
Competitions:	202
Most Recent Event:	Member SRM 482 09.15.10

Division I Submissions

Problem	Submitted	Failed Challenge	Failed Sys. Test	Success %
Level One	202	8	13	89.60%
Level Two	170	15	19	80.00%
Level Three	120	8	29	69.17%
Total	492	31	61	81.30%

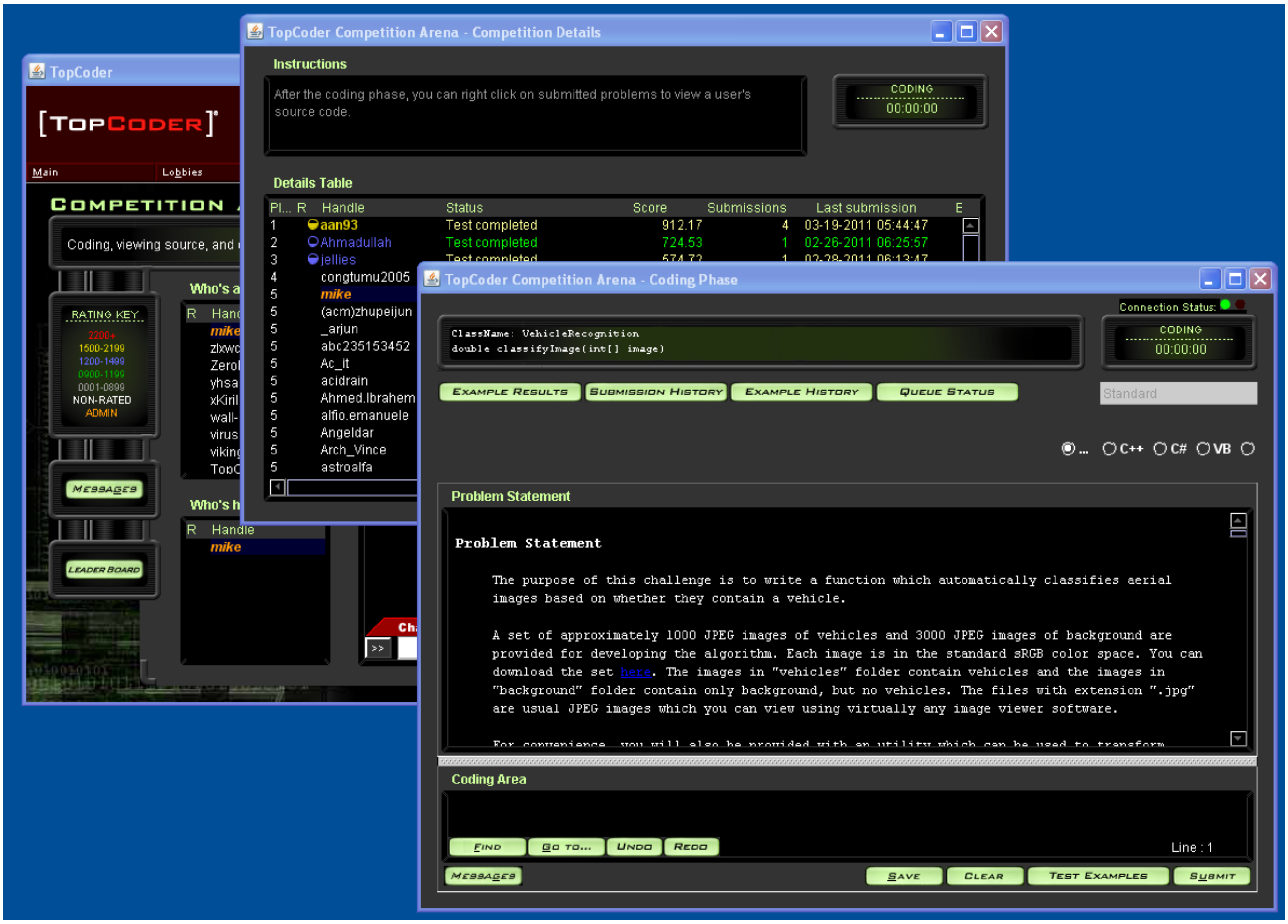
Division II Submissions

Problem	Submitted	Failed Challenge	Failed Sys. Test	Success %
Level One	1	0	0	100.00%
Level Two	1	0	0	100.00%
Level Three	1	0	0	100.00%
Total	3	0	0	100.00%

Challenges

Problem	# Failed Challenges	# Challenges	Success %
Level One	28	74	62.16%
Level Two	53	105	49.52%
Level Three	31	62	50.00%
Total	112	241	53.53%

Appendix 2: TopCoder Participant Arena Screen Shots



APPENDIX 3: Problem Statement

Problem: SpaceMedkit

You have been asked to assist the space medicine community in stocking a space vehicle with appropriate medical resources to mitigate the likelihood for medical evacuation of crew members during space flights. The space vehicle has mass and volume constraints that limit the amount of medical resources that can be flown. To complete this task, you have agreed to create an optimization algorithm that identifies the best possible medical kit (medkit) for meeting constraints on the number of crew member evacuations (**P**) while minimizing the medical resource mass and volume.

For your optimization, the space medicine community will provide you with a list of approved medical resources, with unit mass and volume. Medical resources in the list will be classified as consumable or non-consumable. Consumable resources can only be used once, while non-consumables can be used multiple times.

In order to build the optimization, you will be provided with data from a previously developed mission simulation. Each trial in the simulation provides data for a fully treated (all required medical resources are available), and an untreated scenario (not all required medical resources are available), including the occurrence of a crewmember evacuation. In the simulation, full treatment of a condition does not always prevent evacuation, but it does generally lower the probability of evacuation.

Inputs

The parameters described below will be constant for all tests, and are also available for download. The only parameters that will vary between tests are **P** and **C**.

1. **availableResources** -- this parameter will give you the different medical resources that you may include in your medkit. Each element will be formatted as "RID CONSUMABLE MASS VOLUME".
    - RID is an alphanumeric identifier specific to the resource.
    - CONSUMABLE is either 0 or 1, where 1 indicates that the resource will be used up in treatment (like a drug, for instance) and 0 indicates that the resource can be reused (like a thermometer).
    - MASS and VOLUME are self-explanatory
2. **requiredResources** -- this parameter will describe the different medical events that might occur on the missions. Each event can take one of two courses: a best case course, and a worst case course. These two courses require different resources for treatment. For simplicity, there is no middle ground; the event will follow one of these two courses. Each element of this parameter will be formatted as "MID RID BEST WORST".
    - MID is an alphanumeric identifier specific to the medical event. Note that multiple elements will have the same MID.
    - RID is the resource ID (matching the previously described input)
    - BEST is the amount of this type of resource required if the event takes the best course
    - WORST is the amount of this type of resource required if the event takes the worst course

    (MID,RID) is a unique key for this input, and thus no two elements will have the same value for both of these fields.

3. **missions** -- this parameter will describe a number of missions. Your goal is to design your medkit tailored to these missions. This input should be considered the training data, as your medkit will be evaluated on a different set of missions, which were generated via the same

simulation. Each element will be formatted as "MISSION ORDER MID WORST TREATED UNTREATED".

- MISSION is an id number for the mission
- ORDER specifies the order within a mission that events occur (each mission will be sorted by this in the input)
- MID is an alphanumeric identifier specific to the medical event.
- WORST is 1 if the worst case course of this event occurred, and 0 otherwise (best case)
- TREATED specifies the number of evacuations if this event is treated
- UNTREATED specifies the number of evacuations if this event is untreated

Output

You should design a medkit and return a String[] where each element is formatted as "RID QUANTITY", indicating that the resource QUANTITY of RID should be included (this may be a floating point value).

Your return will be evaluated on each mission independently (resources are restocked between missions). For a mission, the events will be evaluated one by one (according to ORDER). If all of the resources are available to treat the event (under the condition -- best or worst -- that occurs), those resources will be used to treat it. The number of evacuations from the event for the treatment status that occurs will be added to the total number of evacuations. Note that, for simplicity, each medical event is considered independent of the outcome of previous events. This total will be evaluated over all missions. In pseudocode:

```
foreach mission     restock resources according to your output     foreach event in
mission (in order)         if all resources available to treat event
evacuations += event.treated             decrement consumed resources         else
evacuations += event.untreated
```

Scoring

For each test case, your input will be evaluated on a set of 10,000 missions, randomly selected from a corpus of 200,000. The average number of evacuations per mission must be no more than the input **P**. Thus the total number of evacuations summed over all missions must be no more than **P*10000**. Given that, your score will be 1000 / (mass + **C** * volume), where **C** is an input parameter. If the evacuations rate exceeds **P**, your score will be 0 for that test case. Your overall score will be the sum of your individual scores.

Appendix 4: Eliciting Preferences

Choice Survey Email Communication:

[TopCoder]

Subject: Mandatory Survey - NASA-TopCoder Challenge – Please respond in 24 hours

Dear <Handle Name>,

We are considering you as one of the participants for next week's TopCoder-NASA Marathon Match Challenge. We would like you to complete a short three question survey regarding the contest. Please complete the survey within 24 hours.

We appreciate your attention to this. Given the experimental nature of this event, we require you not disclose the existence of these questions through personal communications, email, blog postings, forum postings or any other means---or risk disqualification.

Thank for your help and cooperation!

Best,
Mike Lydon
Chief Technology Officer

Please proceed to the following link: <insert link>

Survey Questions:

a) Version 1

Q1 As you know, we are investigating new ways of participating in TopCoder experiments. Some people will be able to work in teams.

Might you be interested in joining a team to compete against other teams?

I DEFINITELY would prefer to join a team
I MIGHT prefer to join a team
I am indifferent or I am not sure
I MIGHT prefer to compete on my own
I DEFINITELY would prefer to compete on my own

Q2 As further clarification, both teams and individual competitors will be in groups of 20 (4x5-person teams or 20 individuals). There will be 5 cash prizes awarded in each

group, either to the winning team or each of the top 5 individuals. So, the chances of winning---in terms of the prizes per each group of 20 people--are the same for both individual and group formats.

Team members will be free to share ideas and code with one-another over a private discussion board. The team will be evaluated as a group, with the best submission of the group representing the group's final submission.

Please confirm or adjust your previous answer:

I DEFINITELY would prefer to join a team
I MIGHT prefer to join a team
I am indifferent or I am not sure
I MIGHT prefer to compete on my own
I DEFINITELY would prefer to compete on my own

Q3 Finally, here is a hypothetical question. Imagine if TopCoder were always to offer the options of joining a team or competing on your own. What is the best guess of the percentage of events that you would join a team:

I would always join teams (100% of the time in teams)
I would mostly join teams (>80% of the time in teams)
I would frequently join teams (60%-80% of the time in teams)
I would join both roughly equally (40%-60% of the time in teams)
I would somewhat regularly join teams (20%-40% of the time in teams)
I would occasionally join teams (<20% of the time in teams)
I would never join teams (0% of the time in teams)

As a reminder please do not disclose this survey to anyone else.

b) Version 2

Q1 As you know, we are investigating new ways of participating in TopCoder experiments. Some people will be able to work in teams.

Might you be interested in joining a team to compete against other teams?

I DEFINITELY would prefer to compete on my own
I MIGHT prefer to compete on my own
I am indifferent or I am not sure
I MIGHT prefer to join a team
I DEFINITELY would prefer to join a team

Q2 As further clarification, both teams and individual competitors will be in groups of 20 (4x5-person teams or 20 individuals). There will be 5 cash prizes awarded in each group, either to the winning team or each of the top 5 individuals. So, the chances of winning---in terms of the prizes per each group of 20 people--are the same for both individual and group formats.

Team members will be free to share ideas and code with one-another over a private discussion board. The team will be evaluated as a group, with the best submission of the group representing the group's final submission.

Please confirm or adjust your previous answer:
I DEFINITELY would prefer to compete on my own
I MIGHT prefer to compete on my own
I am indifferent or I am not sure
I MIGHT prefer to join a team
I DEFINITELY would prefer to join a team

Q3 Finally, here is a hypothetical question. Imagine if TopCoder were always to offer the options of joining a team or competing on your own. What is the best guess of the percentage of events that you would join a team:

I would never join teams (0% of the time in teams)
I would occasionally join teams (<20% of the time in teams)
I would somewhat regularly join teams (20%-40% of the time in teams)
I would join both roughly equally (40%-60% of the time in teams)
I would frequently join teams (60%-80% of the time in teams)
I would mostly join teams (>80% of the time in teams)
I would always join teams (100% of the time in teams)

As a reminder please do not disclose this survey to anyone else.