

THE PROGRAMMING PERFORMANCE PROPHECIES: PREDICTING STUDENT ACHIEVEMENT IN A FIRST-YEAR INTRODUCTORY PROGRAMMING COURSE

Jeff Ringenberg, Marcial Lapp, Apoorva Bansal, Parth Shah
Electrical Engineering and Computer Science
University of Michigan

Abstract

Ensuring student success in first-year introductory programming courses presents a unique challenge when considering the diversity of student educational backgrounds. Some students enter college having programmed for several years, while others have had little to no exposure to basic programming concepts. In addition, students frequently possess a wide range of skills in areas of study that are related to programming. When students with widely differing skill sets are enrolled in the same introductory programming course, a competency imbalance is immediately established that negatively impacts the course and leads to high attrition rates among students with less experience.

In order to create a more equitable experience and to ensure that students are placed in a course whose difficulty is commensurate with their abilities, we present a methodology to predict student performance in first-year introductory programming courses. Our technique requires that all students take an online exam at the beginning of the term consisting of a variety of algorithmic, logic, and math-based questions that are not tied to a specific programming language. From a sample of over 800 students, we show a significant correlation between the performance on our exam and overall course performance, thereby giving us the ability to properly accommodate students based upon our expectation of their performance in the course and to minimize any potential imbalances that may occur.

Introduction

Introductory programming courses at many universities typically include students with a wide variety of computing backgrounds. Some students may arrive with many years of programming experience from courses they took in high school while others may only be able to do basic computing tasks such as word processing. This often creates a disparity in the classroom and the students with little background in the subject resent the fact that they are competing with students who have more experience. In addition, those students that do have experience tend to feel that the pace of the course is too slow and that they are being held behind.

In order to address this difference in backgrounds, this paper proposes a technique to help identify those students who may have a natural proclivity for programming or who may already have experience in the area. In addition, the technique can assist in finding those students who may struggle with the material more than normal. If these groups of students can be identified, students can be separated into different sections of the course that cover topics at a level of detail and a rate of speed that is commensurate with their abilities.

The specific technique in this paper that is proposed to measure student abilities takes the form of an online exam composed of algorithmic, math, and logic-based questions. The exam is given early in the semester before the students have received any formal instruction in programming. It requires little overhead and is very easy to use. Results from each student are gathered very quickly and can

provide an early estimate of student performance in the course.

The rest of this paper is organized as follows: The next section discusses previous work that has been done in the area. Then we describe our methodology, and present our results. This is followed by some concluding remarks and a discussion of future work. Finally, we conclude with two appendices that list the questions that were used by two different offerings of the exam.

Previous Work

Many attempts have been made to predict student performance in computer programming courses. Factors such as a student's high school GPA, ACT/SAT scores, personality traits, mathematical aptitude, prior exposure to specific courses, and many other points of information have been studied, with varying degrees of success, to see if they are correlated to programming performance. Some correlations have been found to exist, however, much of this data can be very hard to obtain and the processes involved in their analysis can be very work intensive.

Beginning in the early 1980s, Butcher[1] and Muth looked at high school records and ACT scores to measure their predictive abilities. Their results showed that ACT math scores and high school GPA were highly correlated to a student's performance in an early computer science course and that other factors like the student's high school rank and class size were not correlated. Dey[2] and Mand focused their study specifically on a student's math GPA and found that it was also correlated. Additional work done in this area by Campell[3], Cantwell-Wilson[4] and Evans[5] has further corroborated these results.

Dehnadi[6] used a different approach and attempted to show that a student's mental model, as applied to variable assignments in programming, could predict performance.

However, Caspersen[7] et. al. showed that this was clearly not true when attempting to reproduce the results. They offer a few reasons as to why this may have been the case, but ultimately the conclusion is made that predicting performance in a computer science course based on non-performance related data is very difficult.

Bennedsen et. al. [8,9] further studied the effects of non-performance based metrics and found that there is no significant correlation between a student's cognitive development and programming ability. In particular, a student's ability to think abstractly in a general sense cannot be used to predict his/her performance in a programming course. They note, however, that this does not mean that abstraction is not important. In fact, the opposite is the case. Unfortunately, until students are taught how abstraction applies specifically to programming, the general application of abstract thinking skills is not enough to measure a student's natural ability to excel at programming.

Methodology

Testing Environment and Format

To facilitate the easy distribution and collection of the exam results, students took the exam using an online testing tool hosted inside a standard web browser. Only those students enrolled in the course were given access to the exam. Individual exam results were automatically tabulated and sent to the course staff once each exam was completed. In addition, if a student encountered a technical difficulty with the exam that prevented him/her from completing it, s/he was given an opportunity to take it again and each attempt was logged separately. Fortunately, this only happened to a small number of students and there was no effect on the overall exam results.

Testing Sample

The exam was initially given to a group of approximately 350 students in first-year

introductory computer programming courses at the University of Michigan during the Fall 2009 semester. Each student was given 30 minutes to complete the exam that consisted of 25 multiple-choice questions intended to test his/her core, prerequisite knowledge of the course material. The exam was not proctored and the students were expected to take it at any point over the course of 3 days. Over the following semester, the results of the Fall 2009 exam were analyzed to create a new set of questions. Then, during the Fall 2010 semester, a new exam was created using this new set of questions and was given to a group of approximately 450 students in first-year introductory programming courses. The Fall 2010 exam differed from the Fall 2009 exam in that it contained only 16 questions, the questions were both multiple-choice and free-response, and the students were given 45 minutes to complete it.

In order to prevent the students from working with other students on the exam, they were informed that their scores did not affect their course grade and that only their participation was required. Additionally, the time constraint was used to help prevent this type of collaboration since it meant that the students would not be able to consult with each other if there was ever any disagreement on an answer. In other words, once the student figured out an answer, s/he needed to use it and immediately move on to the next question. Incidentally, this wouldn't have prevented a student from having someone else take his/her exam, however, it is the best possible way to deliver the exam without requiring a formal proctoring procedure. Finally, the order in which the questions were presented was randomized so that students could not easily make an answer key to distribute to their classmates.

Question Selection

Questions on the exam required no prior programming knowledge. Instead, they attempted to measure the core knowledge that is required to excel in computer programming

without the student ever having been exposed to programming. Questions were broken up into three different categories: math, logic, and algorithmic. Math questions were intended to explore a student's mathematical abilities using basic algebra questions. The logic and algorithmic questions were meant to focus on the more programming-specific types of knowledge that would help a student to readily grasp such programming concepts as selection and iteration.

Programming questions tied to a specific programming language were not included since some students may have had a natural gift in the area and others may have had experience in many different languages, all of which could not be tested by a single exam. Without tying a student's abilities to a specific programming language, any bias introduced by a student's syntactic knowledge of the language was removed and the exam was better able to determine if the student should take a non-standard version of the course, whether it was more advanced or rudimentary.

Results and Possible Extensions

To determine and refine the predictive abilities of our exam, we first correlated the Fall 2009 predictive exam scores to student performance in a computer programming course using normal exam scores and overall course grades. The results of this initial analysis are shown in Table 1. From the results, there appears to be a definite relationship between a student's performance on the predictive exam and their performance in the course.

Exam 1	0.38
Exam 2	0.40
Exam 3	0.37
Exam 4	0.35
Overall Course Grade	0.33

Table 1: Correlation Statistics for Fall 2009.

The results from Fall 2009 were then analyzed during the following semester to identify those

questions that were the most effective at predicting overall performance. This analysis used an ad hoc correlation and refinement technique since a per-question correlation analysis was too fine-grained to produce useful results. To begin, all questions that over 80% of the students answered correctly were ignored. These questions were considered *ineffective* since it is unlikely that they would be able to differentiate one student from another. Next, sets of questions containing 7 questions each were created by choosing all possible combinations of 7 *effective* questions (i.e. those questions that less than or equal to 80% of the class got correct). Each set of 7 questions was then correlated to overall course grades. Sets with a correlation of less than 0.4 were marked *ineffective* and were no longer considered. Finally, the number of times that each *effective* question appeared in an *effective* set was calculated as a percentage.

The results of this analysis appear in Figure 1 where the bars represent the percentage of the

effective sets in which each effective question appeared. Also included in Figure 1 is a line that indicates the percentage of the students that got each question correct. The questions that exhibit the best predictive abilities are those that have a large value for the bar indicating that they were in many of the "predictive" sets. A smaller value for the line indicating that not many students got the question correct is also desirable, but it is less important. Appendix A lists the questions that were used for the Fall 2009 exam.

Using the results of the analysis on the Fall 2009 data, a new set of 16 multiple-choice and free-response questions was developed for use in the upcoming Fall 2010 semester. This new set of questions included not only the questions from the Fall 2009 exam that were deemed most effective (i.e. questions 5, 6, 12, 19 and 22), but also a new set of related questions that were inspired by all the effective questions from Fall 2009.

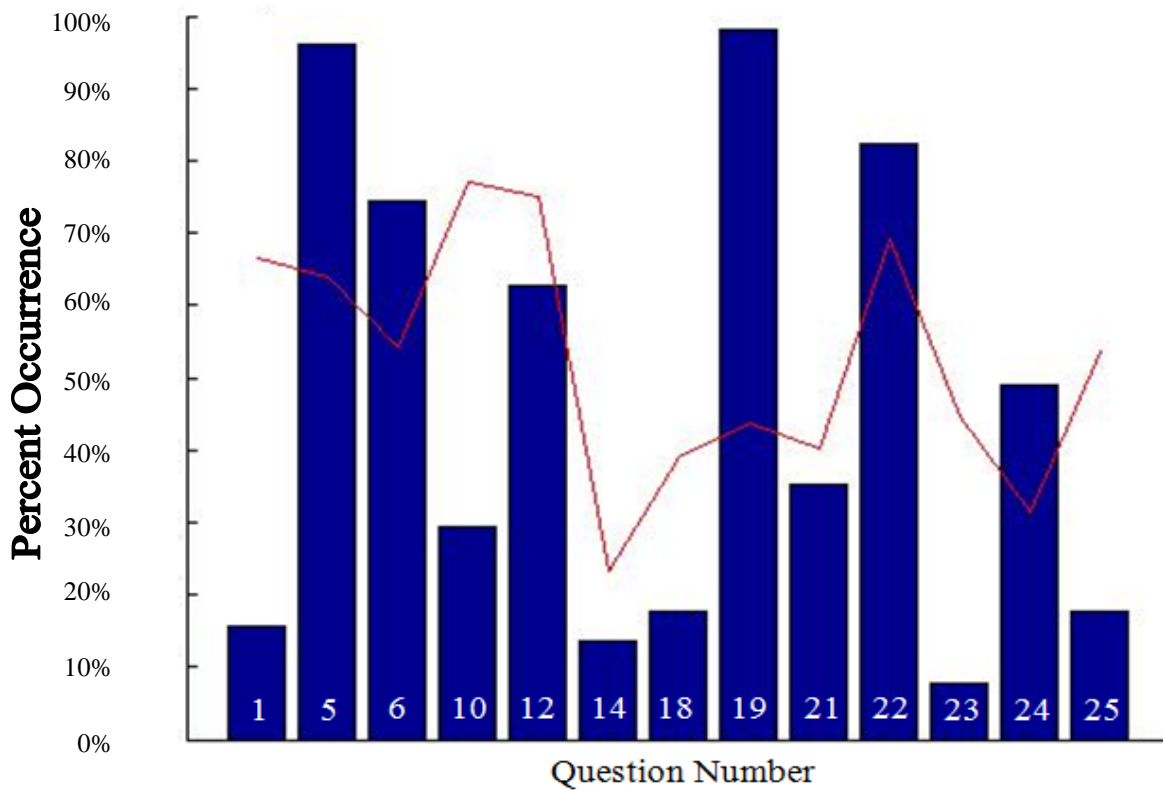


Figure 1: Ad Hoc Correlation Analysis for Fall 2009 Predictive Exam.

During the Fall 2010 semester, two different versions of the introductory programming course were offered. One version of the course, referred to as the *regular* version, moved at an average pace and was taken by the vast majority of the students. The second version of the course, referred to as the *accelerated* version, moved at a much quicker pace and covered a great deal more material. Approximately 365 students took the regular version of the course and 85 took the accelerated version. Students self-selected whether or not to take the regular version or the accelerated version based on personal preference, advice from academic advisors, and the results of an orientation survey that was created using the questions from the Fall 2009 predictive exam as reference.

The outcomes of the Fall 2010 predictive exam were then correlated to overall class performance in both versions of the course. The results for the regular version of the course can be seen in Table 2 and the accelerated version in Table 3. It should be noted that the Fall 2010 semester did not contain a fourth exam.

Exam 1	0.36
Exam 2	0.42
Exam 3	0.36
Overall Course Grade	0.39

Table 2: Correlation Statistics for the Fall 2010 offering of the *regular* course.

Exam 1	0.49
Exam 2	0.43
Exam 3	0.51
Overall Course Grade	0.48

Table 3: Correlation Statistics for the Fall 2010 offering of the *accelerated* course.

As Table 1 and 2 show, the overall correlation to regular exam scores for the regular course remained roughly the same. However, the correlation of the predictive exam to the overall course grade *increased* from 0.33 to 0.39. In addition, the results in Table 3 show that the accelerated course had an even greater correlation of 0.48. This shows that the question refinement technique discussed previously, along with the addition of new and related questions, was effective in increasing the

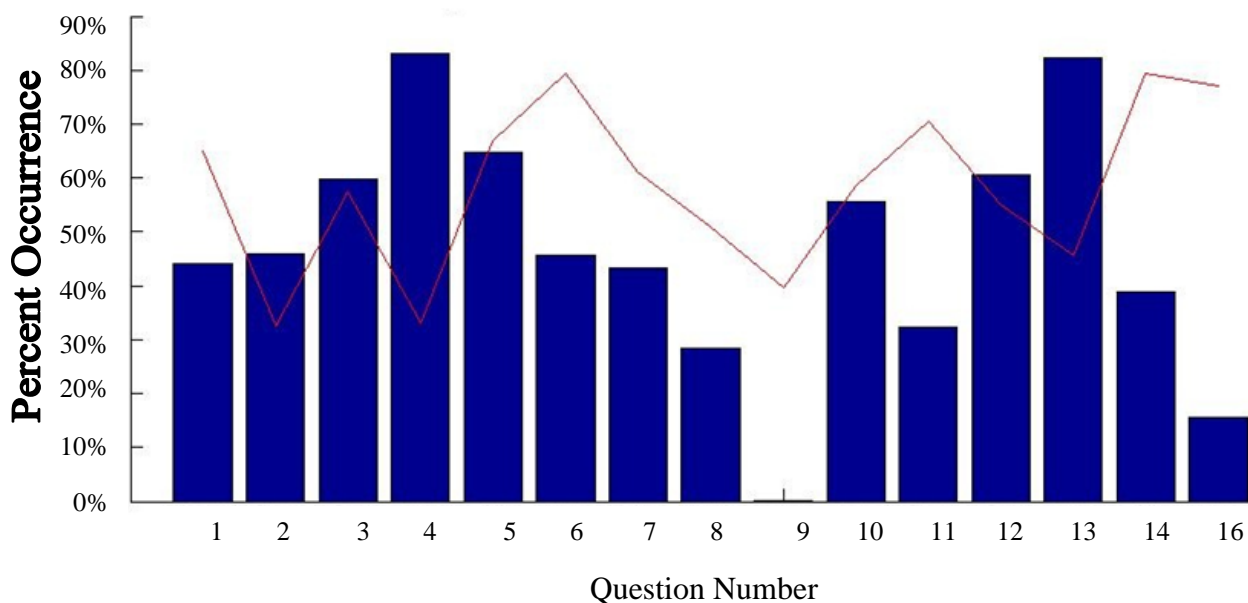


Figure 2: Ad Hoc Correlation Analysis for Fall 2010 Predictive Exam.

predictive abilities of the exam. A final analysis was done using the same ad hoc technique that was used with the Fall 2009 data. The results of this were combined for both versions of the course and are presented in Figure 2. Appendix B lists the questions that were used for the Fall 2010 exam.

As Figure 2 shows, the individual questions for the Fall 2010 semester showed much better predictive ability than the questions that were used during the Fall 2009 semester. In addition, the number of questions that were deemed effective was much higher for Fall 2010 than for Fall 2009. Overall, 15 of the 16 questions for Fall 2010 were marked effective compared to only 13 of the 25 questions from Fall 2009. This further shows that the question refinement technique that was used to generate the questions for the Fall 2010 predictive exam was effective and it produced an exam that was more efficient and better at predicting overall performance in the course. However, one result that showed that the question refinement technique may not be completely accurate is the fact that question 15 from the Fall 2010 exam was marked as ineffective even though it was one of most predictive questions from the Fall 2009 exam.

In addition to looking at correlation data, the median and average scores for the Fall 2010 predictive exam were calculated for both the regular and accelerated versions of the course. The results of this analysis can be seen in Table 4 where all scores are out of a total of 16 points.

Regular Version: Median	9	56%
Accelerated Version: Median	12	75%
Regular Version: Average	9.3	58%
Accelerated Version: Average	11.8	74%

Table 4: Median and Average Exam Scores for Fall 2010.

As the results in Table 4 show, the students in the accelerated version of course performed significantly better with a median score that was 19% higher. This indicates that the current

method for deciding which version of the course that a student should take based on preference, counseling, and the results of the orientation survey may be effective in helping a student pick a version of the course that is commensurate with his/her abilities.

As a possible extension to the current version of the predictive exam, it would be possible to incorporate additional information that could easily be gathered about each student into the analysis. That is, each student could be asked three additional questions to gain a more in depth understanding of his/her previous exposure to computer programming. These questions would be as follows:

1. How much do you know about programming?
 - a. A lot
 - b. A little
 - c. Nothing
2. Have you ever taken a programming course?
 - a. Yes, I took a C, C++, or JAVA course.
 - b. Yes, I took a Visual Basic course.
 - c. Yes, I took a FORTRAN course.
 - d. Yes, I took a course in another programming language.
 - e. No, I did not.
3. Have you ever taken a logic course?
 - a. Yes
 - b. No

These questions can potentially further influence the effects of the predictive exam. One might hypothesize that students who have taken programming courses in the past will fare better in introductory programming courses. Such a relationship has been suggested by Hagan et al.[10]

Conclusions and Future Work

As our results have shown, student performance on the proposed predictive exam is correlated to overall exam performance and that it is possible to refine the questions that are used to chose those that are more predictive than

others. Using this knowledge, it may be possible to predict student performance in an introductory programming course and help students decide the difficulty level of the course that they should take when they first enter college. When combined with other information such as high school GPA and SAT/ACT performance, this information has the potential to greatly impact a student's initial experiences with programming at the college level by placing the student in a course that matches his/her potential.

While our results do show promise, there are definitely many areas of improvement. Since we have only given the exam twice, later offerings of the course can continue to tailor the questions based upon the results that we have gathered. In addition, further studies of the questions using different statistical techniques should be done to decide which questions are more appropriate than others. Finally, questions that measure student perceptions could also be included to add a different dimension to the exam that would further strengthen the results.

Bibliography

1. Butcher, D.F., & Muth, W.A. (1985). Predicting performance in an introductory computer science course. *Communications of the ACM*, 28, 263-268.
2. Dey S. & Mand L. (1986). Effects of mathematics preparation and prior language exposure on perceived performance in introductory computer science courses. In *Proceedings of the 17th SIGCSE Symposium on Computer Science Education*, 144-148.
3. Campbell P. F., & McCabe, G. P. (1984). Predicting the success of freshmen in a computer science major. *Communications of the ACM*, 27(11):1108–1113.
4. Cantwell-Wilson B. & Shrock, S. (2001). Contributing to success in an introductory computer science course: a study of twelve factors. In *Proceedings of the thirty-second SIGCSE technical symposium on Computer Science Education*, 184–188.

5. Evans, G. E. & Simkin, M. G. (1989). What best predicts computer proficiency? *Communications of the ACM*, 32(11):1322–1327.
6. Dehnadi S. & Bornat R. (2006). The camel has two humps. www.cs.mdx.ac.uk/research/PhDArea/saeed/paper1.pdf
7. Caspersen M., Larsen K., and Bennedsen J. (2007). Mental models and programming aptitude. *Proceedings of the 12th annual SIGCSE Conference on Innovation and Technology in Computer Science Education*, 206-210.
8. Bennedsen J. & Caspersen M. (2006). Abstraction ability as an indicator of success for learning object-oriented programming? *ACM SIGCSE Bulletin*, 38(2): 39-43.
9. Bennedsen J. & Caspersen M. (2008). Abstraction ability as an indicator of success for learning computing science? *Proceedings of the Fourth international Workshop on Computing Education Research*, 15-26.
10. Hagan, D. & Markham, S. (2000). Does it help to have some programming experience before beginning a computing degree program? In *Proceedings of the 5th annual SIGCSE/SIGCUE Conference on Innovation and Technology in Computer Science Education*, 25–28.

Biographical Information

Jeff Ringenberg is a Lecturer at the University of Michigan's College of Engineering. His research interests include mobile learning software development, tactile programming, methods for bringing technology into the classroom, and studying the effects of social networking and collaboration on learning. He holds BSE, MSE, and PhD degrees in Computer Engineering from the University of Michigan.

Marcial Lapp is a graduate student in the Industrial and Operations Engineering Department at the University of Michigan. His research interests lie in modeling and solving large-scale optimization problems focused on the transportation and logistics industries. He holds a PhD in Industrial and

Operations Engineering, and MBA in Finance and Corporate Strategy, and Masters and Bachelors degrees in Computer Science from the University of Michigan.

Apoorva Bansal holds BSE degrees in Computer Engineering and Mechanical Engineering from the

University of Michigan and is currently continuing in graduate school at the University of Michigan.

Parth Shah holds a BSE in Computer Science Engineering from the University of Michigan and currently works at Microsoft.

Appendix A - Questions Used for the Fall 2009 Predictive Exam.

Question Number	Question Text	Possible Answers	Correct Answer
1	The average (arithmetic mean) of x and y is 20. If $z=5$, what is the average of x , y , and z ?	a) $8 \frac{1}{3}$ b) 10 c) 15 d) $17 \frac{1}{2}$	c
2	If $3x - 2 = 7$, then $4x = ?$	a) 3 b) 5 c) 9 d) 12	d
3	If $x = 7$ and later y is assigned the value of $x - 5$, then finally $9x = ?$	a) 18 b) 3 c) 9 d) 63	d
4	If $x = 10$, then at a later point in time x 's value becomes $2*x + 1$, so finally $x = ?$	a) 10 b) 20 c) 2 d) 21 e) -2	d
5	To reproduce an old photograph, a photographer charges x dollars to make a negative, $(3x)/5$ dollars for each of the first 10 prints, and $(x/5)$ dollars for each print in excess of 10 prints. If \$45 is the total charge to make a negative and 20 prints from an old photograph, what is the value of x ?	a) 3 b) 3.5 c) 4 d) 4.5 e) 5	e
6	If $0 < s*t < 1$, then which of the following can be true?	a) $s < -1$ and $t > 0$ b) $s < -1$ and $t < -1$ c) $s > -1$ and $t < -1$ d) $s > 1$ and $t < -1$	c
7	In a certain shop, notebooks that normally sell for 59 cents each are on sale at 2 for 99 cents. How much can be saved by purchasing 10 of these notebooks at the sale price?	a) \$0.85 b) \$0.95 c) \$1.10 d) \$2.00	b
8	If the average (arithmetic mean) of 5 consecutive integers is 12, what is the sum of the least and greatest of the 5 integers?	a) 24 b) 14 c) 12 d) 11	a
9	The cost, in dollars, of manufacturing x refrigerators is $9,000 + 400x$. The amount received when selling these x refrigerators is $500x$ dollars. What is the least number of refrigerators that must be manufactured and sold so that the amount received is at least equal to the manufacturing cost?	a) 10 b) 18 c) 45 d) 90 e) 100	d

10	In a rectangular coordinate system, what is the area of a triangle whose vertices have the coordinates (3,0) (5,3) and (5,-3)?	a) 3 b) 4.5 c) 6 d) 7.5	c
11	If Henry Ford's Rouge River factory turns out 10 Model T's in 18 minutes, approximately how many could it produce in one hour?	a) 27 b) 28 c) 30 d) 33	d
12	A ranch has both horses and ponies. Exactly $\frac{5}{6}$ of the ponies have horseshoes and exactly half of the ponies with horseshoes are Icelandic ponies. What is the minimum number of ponies on the ranch?	a) 6 b) 8 c) 12 d) 18	c
13	If n is an even integer, which of the following must be an odd integer?	a) $3n-2$ b) $3(n+1)$ c) $n-2$ d) $\frac{n}{3}$	b
14	The first three terms of a sequence are 2, 7, and 22. After the first term, each consecutive term can be obtained by multiplying the previous term by 3 and then adding 1. If the sequence continues to be expanded, what will the sum of the tens digit and the units digit of the thirty-fifth term of the sequence be?	a) 2 b) 4 c) 7 d) 10 e) 12	b
15	If $X = 10$ and $Y = 20$, and then later X is assigned Y 's value, then what do X and Y finally equal?	a) $X = 10, Y = 10$ b) $X = 10, Y = 20$ c) $X = 30, Y = 20$ d) $X = 20, Y = 20$ e) $X = 0, Y = 0$	d
16	If $X = 10$ and $Y = 20$, and then later Y is assigned to X 's value, and even later X is assigned to Y 's value, then what do X and Y finally equal?	a) $X = 10, Y = 10$ b) $X = 10, Y = 20$ c) $X = 20, Y = 30$ d) $X = 20, Y = 20$ e) $X = 0, Y = 0$	a
17	If $X = 1$ and $Y = 3$ and $Z = 5$, and then later X is set to Y 's value, and even later Y 's value is set to the value of Z , then what do X , Y , and Z finally equal?	a) $X=1, Y=1, Z=1$ b) $X=3, Y=3, Z=5$ c) $X=4, Y=8, Z=5$ d) $X=3, Y=5, Z=5$ e) $X=1, Y=3, Z=3$	d
18	Jill was looking at a picture when Jack walked up and asked, "Jill, who are you looking at?". Jill replied, "I have no sisters and this woman's mother is my mother's daughter." Who is Jill looking at?	a) Herself b) Her mother c) Her grandmother d) Her daughter	d
19	Assuming a base 2 numbering system, $00110001 + 10001001 = ?$	a) 10111001 b) 10111010 c) 10111002 d) 00111001	b
20	There was a robbery in which 500 iPods were stolen in 30 seconds. The robber(s) quickly left in a Mini Cooper S. It is known that: 1. Nobody could have committed the crime other than Larry, Moe, and Curly. 2. Curly never commits a crime without Larry also committing the same crime. 3. Moe cannot drive.	True False	b

	True or False: Larry is innocent.		
21	There are three boxes. One is labeled "APPLES", another is labeled "ORANGES", and the last one is labeled "APPLES AND ORANGES". You know that every box is labeled incorrectly. You may ask me to pick one fruit from one box which you choose. From which box should I pick a fruit in order to label all the boxes correctly?	a) APPLES b) ORANGES c) APPLES AND ORANGES d) Either APPLES or ORANGES	c
22	Statement 1: The number of false statements here is one. Statement 2: The number of false statements here is two. Statement 3: The number of false statements here is three. Statement 4: The number of false statements here is four. Assuming that only one of the four above statements is true, which one must it be?	a) Statement 1 b) Statement 2 c) Statement 3 d) Statement 4 e) None of the statements is true	c
23	Consider a robot with the following behavior: 1. When you turn on the robot, it goes outside. 2. If the robot goes outside and is raining, then it goes back inside and takes an umbrella from a bin containing 6 umbrellas. If the bin is empty, the robot continues unaffected. 3. If the robot goes outside and it is not raining, then it drops all its umbrellas on the ground (if it has any) and goes back inside. 4. The robot always takes 1 minute to get outside and then 1 more minute to come back inside. 5. The robot can instantly pick up and drop umbrellas (i.e. it takes no time). 6. After the robot arrives back inside, it begins to go back outside 13 minutes later continuing to do so until it is turned off. Based on the above behavior, if it starts raining at 1:00pm and continues to rain until 2:00pm, how many umbrellas will be on the ground at 3:00pm if you turn on the robot at 1:30pm?	a) 0 b) 1 c) 2 d) 3 e) 4	c
24	Using the robot behavior from Question 23, if it starts raining at 1:59pm and continues to rain until 3:01pm, how many umbrellas will be on the ground at 3:30pm if you turn on the robot at 1:59pm?	a) 1 b) 3 c) 5 d) 6 e) 7	c
25	Using the robot behavior from Question 23, if it starts raining at 1:59pm and continues to rain until 3:01pm, how many umbrellas will be on the ground at 3:30pm if you turn on the robot at 1:59pm?	a) 1 b) 3 c) 6 d) 8 e) 9	c

Appendix B - Questions Used for the Fall 2010 Predictive Exam.

Question Number	Question Text	Possible Answers	Correct Answer
1	<p>What does this list of instructions do?</p> <ol style="list-style-type: none"> 1. Request a value for A 2. Request a value for B 3. Request a value for C 4. Assign the value of the equation $(B \times B - 4 \times A \times C)$ to D 5. Return D 	<ol style="list-style-type: none"> a) It returns the discriminant for a quadratic equation $Ax^2+Bx+C=0$ b) It returns the solution to a quadratic equation $Ax^2+Bx+C=0$ c) It returns the equation Ax^2+Bx+C d) None of the above 	a
2	<p>What does this list of instructions do (assuming that the initial value for A is a positive integer)?</p> <ol style="list-style-type: none"> 1. Request a value for A that is a multiple of 3 2. Assign A plus 1 to A 3. While A is not equal to 0, assign A minus 3 to A 4. Return A 	<ol style="list-style-type: none"> a) It returns a multiple of A plus 1 b) It returns the smallest divisor of A c) It returns the value 0 d) It returns the value 1 e) The instructions will run forever 	e
3	<p>What does this list of instructions do (assuming that the initial values for A and B are positive integers and that instruction 5 will eventually execute)?</p> <ol style="list-style-type: none"> 1. Request a value for A 2. Request a value for B 3. While B is greater than A, request a new value for B 4. Assign B divided by A to C 5. Return C 	<ol style="list-style-type: none"> a) It returns a value less than or equal to 1 b) It returns 0 c) It returns a value greater than or equal to 1 d) It returns 1 e) None of the above 	a
4	<p>What does this list of instructions do (assuming that the initial values for X and Y are positive integers)?</p> <ol style="list-style-type: none"> 1. Request a value for X 2. Request a value for Y 3. While X is greater than or equal to Y, assign X minus Y to X 4. Return X 	<ol style="list-style-type: none"> a) It returns the value X divided by Y b) It returns the integer remainder of the initial value of X divided by the initial value of Y c) It returns the integer remainder of the initial value of Y divided by the initial value of X d) It returns the value of the initial value of Y minus the initial value of X e) The instructions will run forever 	b

5	<p>What does this list of instructions do (assuming that the initial values for A and B are positive integers)?</p> <ol style="list-style-type: none"> 1. Request a value for A 2. Request a value for B 3. If A is less than or equal to B, return B multiplied by 3 4. Otherwise, return A multiplied by 3 	<ol style="list-style-type: none"> a) It returns the maximum of A and B b) It returns the minimum of A and B c) It returns the maximum of A multiplied by 3 and B multiplied by 3 d) It returns the minimum of A multiplied by 3 and B multiplied by 3 	c
6	<p>What does this list of instructions do?</p> <ol style="list-style-type: none"> 1. Request a value for A 2. Request a value for B 3. Assign the value $(A \times A)$ to C 4. Assign the value $(C + B \times B)$ to C 5. Assign the square root of C to C 6. Return C 	<ol style="list-style-type: none"> a) It returns the value $(C + B \times B)$ b) It returns the value $(A \times A + B \times B)$ c) It returns the square root of $(B \times B)$ d) It returns the square root of $(A \times A + B \times B)$ e) None of the above 	d
7	<p>A man decides to buy a horse. He pays 60 dollars for the animal. After a year, the value of the horse has increased to 70 dollars and he decides to sell the horse. A few days later he regrets his decision to sell the horse, and he buys it again. Unfortunately, he has to pay 80 dollars to get it back, thereby paying 10 dollars more for the horse that he just sold! After another year of owning the horse, he finally decides to sell it for 90 dollars. What is the overall profit the man makes? (Please enter only whole numbers and no units or signs such as \$\$ or dollars.)</p>	Free Response	20
8	<p>A cube's exterior surfaces are painted with 3 different colors in such a way that the opposite sides of the cube are painted in the same color. The cube is then cut into 64 smaller cubes of equal size. How many cubes have at least two of their sides painted in different colors?</p>	<ol style="list-style-type: none"> a) 8 b) 16 c) 24 d) 32 e) 56 	d
9	<p>How many squares are there on a normal tic-tac-toe board assuming the board consists of 3 rows and 3 columns? (Please enter only whole numbers and no units such as squares.)</p>	Free Response	14
10	<p>Three people picked 65 apples altogether. At the first tree they each picked the same number of apples. At the second tree they each picked 3 times as many as they picked at the first tree. When they finished at the third tree, the group had 5 times as many apples as they had when they started at that tree. At the fourth tree the group picked just 5 apples. How many apples did each person pick at the first tree?</p>	Free Response	1

11	At a party, everyone shook hands with everybody else. There were 66 handshakes. How many people were at the party? Hint: Consider how many handshakes occur with two people, three people, etc?	a) 8 b) 10 c) 12 d) 14	c
Question 6 from Fall '09	If $0 < s \cdot t < 1$, then which of the following can be true?	a) $s < -1$ and $t > 0$ b) $s < -1$ and $t < -1$ c) $s > 1$ and $t < -1$ d) $s > -1$ and $t < -1$ e) $s > 1$ and $t > 1$	d
Based on Question 19 from Fall '09	Assuming a base 2 numbering system, $00110010 + 10001010 = ?$	a) 10111010 b) 10111020 c) 00111200 d) 10011010 e) 10111100	e
Based on Question 5 from Fall '09	To make money as a painter, an artist must first charge X dollars for an original painting. The artist then charges $(2X)/5$ dollars for each of the first 10 copies of the original painting and finally $(X/10)$ dollars for each additional copy. If the artist made \$280 by selling the original painting along with 30 copies, what was the cost of the original painting?	a) \$30 b) \$35 c) \$40 d) \$45 e) \$50	c
Question 22 from Fall '09	Assuming that only one of the four statements below is true, which one must it be? Statement 1: The number of false statements here is one Statement 2: The number of false statements here is two Statement 3: The number of false statements here is three Statement 4: The number of false statements here is four	a) Statement 4 b) Statement 3 c) Statement 2 d) Statement 1 e) None of the statements is true	b
Question 12 from Fall '09	A ranch has both horses and ponies. Exactly $5/6$ of the ponies have horseshoes and exactly half of the ponies with horseshoes are Icelandic ponies. What is the minimum number of ponies on the ranch?	a) 6 b) 8 c) 10 d) 12 e) 18	d