AN ABSTRACT OF THE THESIS OF

Vaishnavi Narayanan for the degree of Master of Science in Computer Science presented on September 18, 2007.

Title: Gender Differences in End-user Debugging Strategies.

Abstract approved:

____

Margaret M. Burnett

There has been little prior research reporting strategy usage in end-user problem solving, and even less using gender as a factor. Without this type of information, end-user programming systems cannot know the “target” at which to aim, if they are to support male and female end-user programmers’ debugging. As a background to the thesis, an experiment was conducted by our group members, where the participants were given a post session questionnaire that had an open-ended question about what debugging strategies they adopted in finding and fixing errors. It was found that among the mentioned strategies, testing and code inspection had significant statistical differences among male and/or female success groups. This thesis’s goal is the investigation of the behavioral evidence of the two primary strategies, testing and code-inspection using gender as a factor. Using quantitative and qualitative methods, we analyzed the two strategies reported, and looked for relationships among participants’ strategy choices, gender, and debugging success. Our results indicate that males and females debug in quite different ways, and the debugging strategies that worked well for the males were not the same ones that worked well for the females. Our results also reveal that tools currently available to end-user debuggers may be especially deficient in supporting debugging strategies used by females.

Gender Differences in End-user Debugging Strategies

by
Vaishnavi Narayanan

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented September 18, 2007
Commencement June, 2008

Master of Science thesis of Vaishnavi Narayanan presented on September 18, 2007

APPROVED:

Major Professor, representing Electrical and Computer Engineering

Director of the School of Electrical Engineering and Computer Science

Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Vaishnavi Narayanan, Author

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisor, Dr. Margaret Burnett for her guidance. Her enthusiasm has always been a great source of inspiration for me. Throughout my research and thesis-writing period, she provided me with constant support and encouragement without which I would have been lost. I wish to thank her for giving me this great opportunity to work with her.

I would also like to thank Dr. Curtis Cook, Dr. Carlos Jensen and Dr. Richard Poppino for agreeing to be part of my committee. I am particularly grateful to Dr. Cook for his patience and guidance during the initial stages of my research.

I wish to thank my team members, Dr. Laura Beckwith, Neeraja Subrahmaniyan, Valentina Grigoreanu, Joey Lawrance and Flora Tan for their help and feedback. I am thankful to all those involved in the summer 2006 experiment which, provided the necessary background for my thesis.

I also owe many thanks to my friends who have helped me directly or indirectly in my thesis.

Lastly, and most importantly, I am grateful to my family who, despite being thousands of miles away, were always by my side.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

Page

LIST OF FIGURES

LIST OF TABLES

Gender Differences in End-User Debugging Strategies

1. Introduction

Our research group, “Gender HCI”, began investigating gender differences empirically in end-user problem solving environments early in the year 2003, by employing theories from different related domains like psychology etc. This led to many interesting findings regarding females’ self-efficacy and willingness to approach and adopt new features [Beckwith et al. 2005]. Our group also discovered gender differences in playful tinkering with features [Beckwith et al. 2006].

Following this, our group used data-mining as a tool to explore patterns of feature usage by males and females and the relationship with debugging success [Grigoreanu et al. 2006]. This work discovered that there was a similarity in the patterns of successful males and that of *un*successful females. This made us interested in the strategies employed by males and females in problem solving, since the strategies used during the debugging session could have influenced the outcome. We wondered if there was a difference in the feature usage patterns because the strategies employed by males and females in debugging were fundamentally different.

“Strategy” can be defined as a reasoned plan or method for achieving a specific goal.

Gender differences in strategy have received significant research attention in many fields outside problem solving. One such study is in the field of information processing. Meyers-Levy generated a theory called the “Selectivity Hypothesis” to bring together numerous theories of gender differences with respect to information processing [Meyers-Levy 1989]. The theory states that males tend to process information in a manner termed “heuristic”, paying particular attention to cues that are highly available and particularly salient in the focal context. Females, on the other hand, process information in a manner termed “comprehensive”, attempting to assimilate all available cues.

In the area of navigation, there has been research reporting that males and females navigate through the real world using different strategies [Halpern et al. 2000, Kimura et al. 1999]. These findings state that females tend to use landmarks to navigate, while males use the direction in which they are heading. There also exist good summaries of the known gender differences in spatial abilities and navigation strategies [Kimura et al. 1999], and most reports tend to document male advantages in spatial tasks. Several studies suggest that these gender differences are further exaggerated when the spatial task is navigation in a virtual environment [Sandstrom et al. 1998, Waller et al. 1998].

Similarly, in the area of hardware design, it was found that larger displays helped reduce the gender gap in navigating virtual environments [Czerwinski et al. 2002]. With smaller displays, males' performance was better than females'. With larger displays, females' performance improved and males' performance was not negatively affected.

A study of children's problem solving abilities revealed gender differences in strategy use [Fennema et al. 1998]. Girls tended to use concrete modeling (e.g., counting on fingers or counting strategies which were simply a matter of following the methods they were taught), while boys tended to use more abstract strategies such as invented algorithms or derived facts.

In a study of fault localization [Prabhakararao et al. 2004] in the field of end-user problem-solving, users adopted two kinds of strategies namely "ad hoc" and "data flow" when they noticed an incorrect value in the spreadsheet. The dataflow strategy consisted of following the dependencies back from the cell with the incorrect value through cell references until they found the fault. The remaining strategies were termed "ad hoc". The results showed that the dataflow strategy was more successful than ad hoc strategies overall. However, such studies in this field did not take the gender factor into account.

The gender factor is important in designing an end-user problem-solving environment. Ideally, such an environment should be designed to support the strategies of both genders. For this, we first need to understand whether strategies employed by the end users vary based on the gender and if so, which strategies help them in their debugging process. There may be strategies that favor the success of males, but are counterproductive to the success of the females and vice-versa. Studying them will not only help us understand the needs of the end-user programmers during their process but also guide us in designing a gender-neutral environment supporting the needs of both males and females.

Keeping this in mind, members of our group designed an experiment during the summer of 2006, and used it to gather data to investigate the different strategies used by males and females while debugging spreadsheets. Strategy exists only in the head of the subjects and the questionnaire was used to obtain in-the-head data which is not directly observable. One of our group's researchers investigated this further by coding the responses and found that the strategies that had significant statistical differences among male and/or female success groups were: dataflow, testing, and code inspection. Furthermore, testing and code inspection were the most commonly co-occurring among the mentioned strategies and over half of the end-user participants used them together.

Investigating the behavioral evidence of these strategies using qualitative and quantitative methods was a good next step to determine what strategies the participants actually *used* and how it helped in their success. The topic of this thesis is the investigation of the behavioral evidence of those two strategies.

2. Experiment

This chapter describes the experiment conducted on summer 2006 by our group which serves as the background to this thesis. The credit for conducting the experiment and writing this chapter goes to Dr. Margaret Burnett, Neeraja Subrahmaniyan, Valentina Grigoreanu and Dr. Laura Beckwith.

2.1 Participants and Procedures

There were 61 participants: 37 females and 24 males. The participants were undergraduates from a variety of majors. They had prior experience using spreadsheets, but very limited programming experience. Few background differences existed between genders. Those present favored females, who had marginally significantly higher GPAs than the males (males: 3.32 (0.41) females: 3.51 (0.32); ANOVA: $F(1,59)=3.81$, $p<0.06$). Females also had higher academic ages, but not significantly so. Participants' ties to engineering/science/math were very low: only 6 females and 5 males were studying these fields. People studying computer science or who had taken significant computer science coursework were not allowed to participate.

A pre-experiment questionnaire, based on Compeau and Higgins' validated scale [Compeau et al. 1995], contained 10 self-efficacy questions specific to end-user debugging tasks. Bandura's self-efficacy theory [Bandura 1977] defines self-efficacy as a person's belief in his/her ability to do a specific task. There was a significant difference in self-efficacy of males versus females (males: 40.96 (4.87) females: 37.73 (4.93); ANOVA: $F(1,59)=6.30$, $p<0.02$). Some previous studies have also reported lower computer-related self-efficacy scores among females than males (e.g., [Beckwith et al. 2005], [Busch et al. 1995]).

A 25-minute hands-on tutorial (described later in the chapter) was presented to familiarize participants with the spreadsheet features. Subsequently, participants carried out two experimental tasks. Participants' actions and the system's feedback were captured in electronic transcripts, along with their final spreadsheets. A post-session questionnaire assessed participants' comprehension of the features and also asked the participants to describe the strategies they used for finding and fixing errors.

2.2 Software Environment

The environment used in the study is a research spreadsheet environment that includes explicit support for testing and debugging by end-user programmers in the form of WYSIWYT ("What You See Is What You Test"). WYSIWYT is a collection of testing and debugging features for end-user programmers [Burnett et al. 2005]. Although the intent of the WYSIWYT features is to support testing-based strategies, the features were flexible enough to allow participants considerable leeway in the strategies they actually used. We chose to use our research spreadsheet system because its features provide participants more choices of testing and debugging strategies than Excel. Our environment also included a logging capability, which provided the ability to collect the extensive activity data necessary for statistical analysis of behavior patterns.

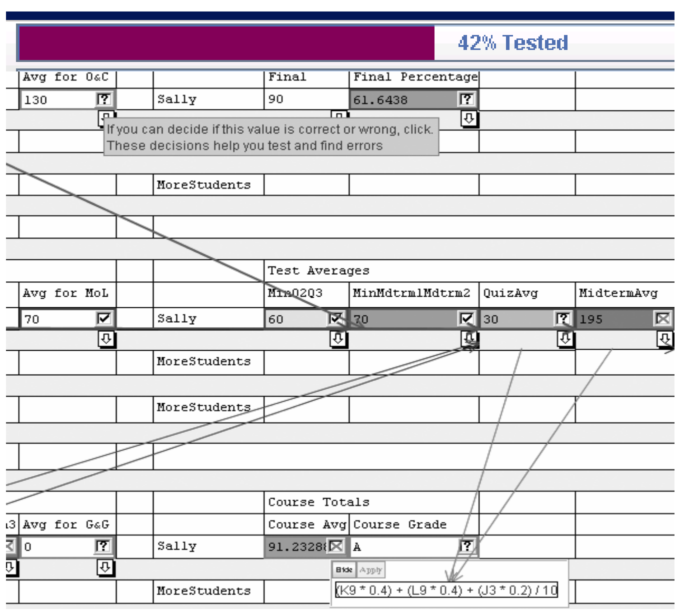


Figure 1. Forms/3 environment (Gradebook spreadsheet).

With WYSIWYT, if the user notices that a cell's value is correct, he or she can check it off. Borders of untested cells are red (light gray in this paper), partially tested cells are a shade of purple (intermediate shades of gray), and fully tested cells are blue (black). Colors reflect how much of the "code" (formula subexpressions) have been covered by the checked-off values. Optional dataflow arrows are also colored to reflect testedness of specific relationships between cells and sub-expressions. For example, if a user checks off the MinMidtrm1Midtrm2 cell in Figure 1, the system updates all affected cell border colors that fed into the answer of MinMidtrm1Midtrm2, the color of any visible dataflow arrows, and a "tested %" progress bar (top of Figure 1), all reflecting the formula expressions covered by the testing so far.

The user might notice that a value is wrong, and can "X it out" (cell Course_Avg in Figure 1). X-marks trigger fault likelihood calculations, which cause interiors of cells suspected of containing faults to be highlighted in shades along a yellow-orange continuum (shades of gray in this paper), with darker orange shades on cells with increased fault likelihood [Burnett et al. 2004].

Sometimes it is not easy to conjure up useful values. In that case, the user can press a "Help Me Test" button (not shown), which suggests values that, if the user checks them off, will enable formula subexpressions to be covered that haven't been covered by prior tests [Burnett et al. 2004].

Each of the features mentioned above is supported through the mechanisms of the Surprise-Explain-Reward strategy. This strategy relies on a user's curiosity about features in the environment. If the user is surprised by something she sees in the spreadsheet, for example, changes in cell border colors, she can then seek out explanations of the features via tooltips [Wilson et al. 2003]; every element in the spreadsheet has an associated tooltip which pops up when the user hovers over it. The aim of the strategy is that, if the user follows up as advised in the explanation, rewards will ensue, such as an increase in testedness progress bar at the top of the spreadsheet [Ruthruff et al. 2004]. The features are also based on the Attention Investment model [Blackwell et al. 2002]. By reading a tool tip, following its advice, and receiving a reward, the user may perceive the value of performing the action to be greater than the costs (in time and effort) of doing it. This encourages further use of features in the spreadsheet.

When the user displays a formula (lower right of Figure 1) by clicking the arrow tab below that cell, it stays displayed until the user closes it. This device allowed participants to have multiple formulas open simultaneously, increasing the viability of debugging strategies based on code inspection, if a participant was so inclined. These and all features in the environment were supported with the tooltips (shaded text box in Figure 1).

2.3 Tutorial

To avoid suggesting strategies to our participants that might introduce bias, the tutorial covered features only. It did not emphasize any particular feature over another, nor did it present any problem-solving scenario that might suggest how to build a strategy using the feature. The participants got explanations of the features and hands-on practice

The tutorial covered six features: Tool tips, Checkmarks, X-Marks, Arrows, Formula Edits, and Help Me Test. Participants also received a one-page quick-reference style handout with all the features, to help them stay oriented in the tutorial and to refer to later in the experiment. At the end of the tutorial, participants were given time to further explore the features by working on a practice spreadsheet debugging task. Half of the tutorial sessions were presented by a male and half by a female. This design ensured that approximately 50% of males were instructed by a same-gender instructor and 50% by an opposite-gender instructor (and likewise for the females) [Whitworth et al. 2002], serving to distribute any gender effect of the tutorial presenter equally over the two genders.

2.4 Tasks and Materials

Participants tested two spreadsheets, Gradebook (Figure 1) and Payroll. Other than the layout, the spreadsheets and the seeded faults were the same as those of [Beckwith et al. 2005a]. While designing the layout, we took care to avoid potential confounds among different sequences participants might follow. For example, Western reading order was distinguishable from description order, and from dataflow order, and so on. The spreadsheets had a total of 11 faults representative of the fault categories in Panko's classification system [Panko 1998], six in Gradebook and five in Payroll.

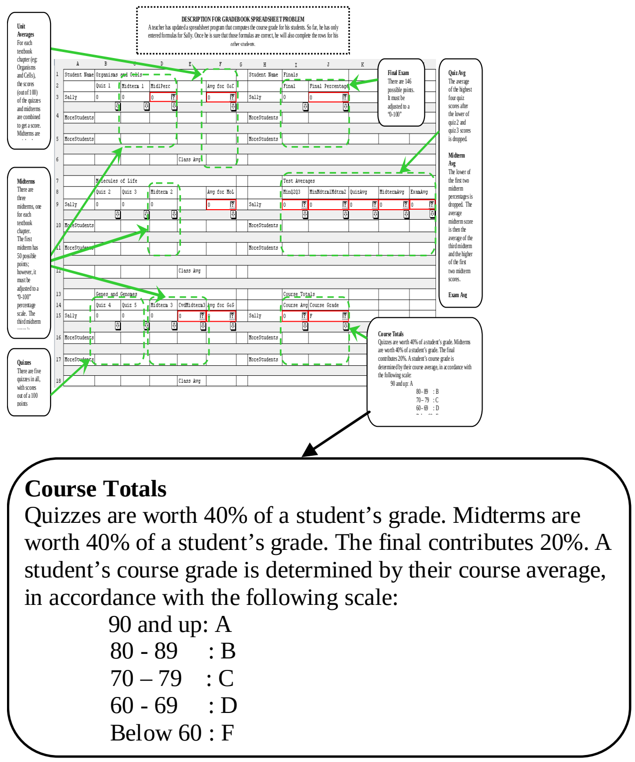


Figure 2. Description handout.

The participants were given the spreadsheet, a handout describing the spreadsheet (Figure 2), and a handout with examples of the spreadsheet with two sets of correct values. The order in which the handouts were collated for the participants was random across tasks, to avoid order exerting any systematic influence on participants' strategy choices. The time limits for the debugging tasks were 22 minutes for Gradebook and 35 minutes for Payroll. The time constraints were meant to simulate time constraints frequently encountered in real world computing tasks and to prevent experimental confounds, such as participants spending too much time on the first task or not enough time on the second task, participants leaving early, etc. The tasks were counterbalanced.

The participants were told that a spreadsheet had been updated and that, “Your task is to test the updated spreadsheet and if you find any errors, fix them.”

3. What Were Successful Females Doing? Code Inspection

3.1 What is Code Inspection?

In the software engineering field, code inspection is defined as a process of examining the computer source code to uncover errors and defects [Pressman et al]. In the Forms/3 environment, the computer source code refers to the formulas and the errors refer to the bugs in the spreadsheet. Hence, code inspection in this environment can be defined as looking at the formulas in order to judge their correctness. This judging is used for deciding if there are any bugs at all and for narrowing down the bug or finding where the bug lies.

Forms/3 allows multiple formulas to be simultaneously displayed (Figure 3) thereby supporting this strategy to some extent. For example, in a code inspection scenario, a participant displays one formula, looks at the code and proceeds to look at another formula. One behavior to suggest code inspection is having multiple formulas displayed simultaneously without editing them, suggesting that their only purpose for being displayed is inspection. In the Forms/3 environment's logging, such behavior would have consecutive POST actions without any formula editing.

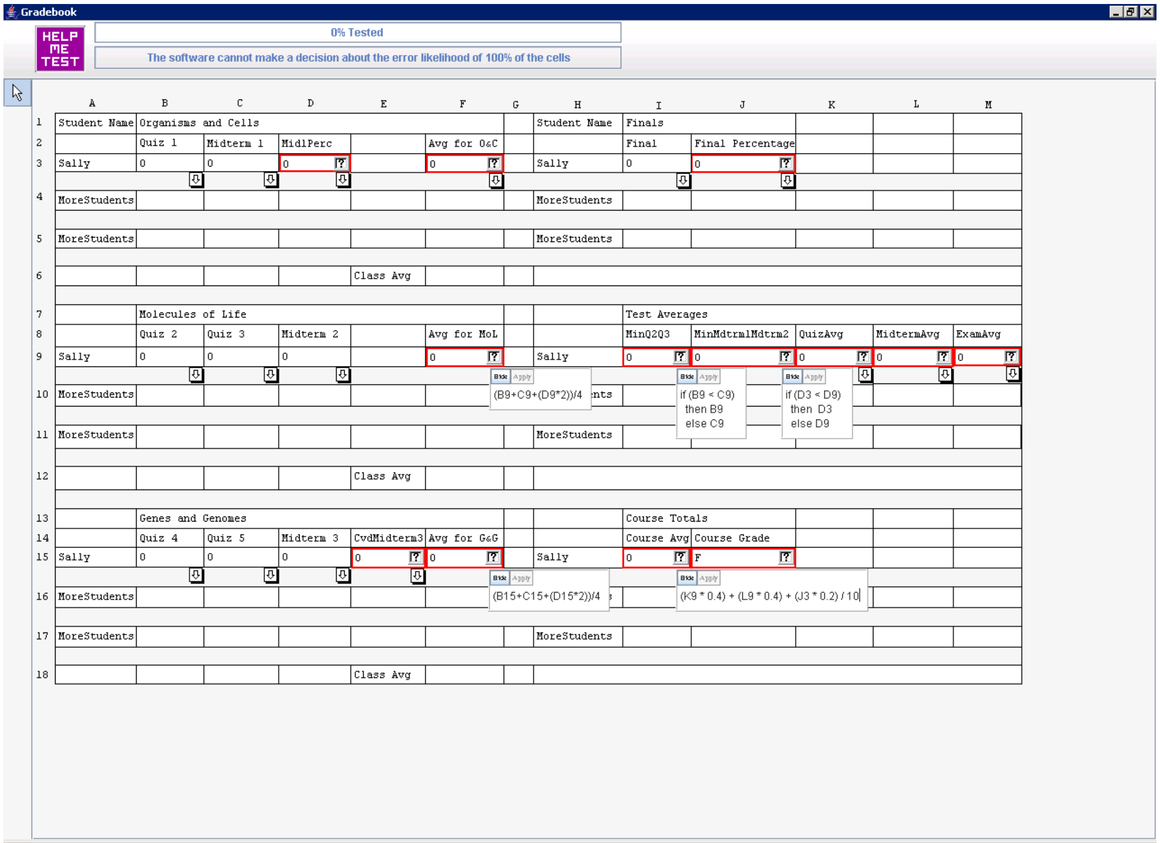


Figure 3. A spreadsheet sample indicating evidence of code inspection.

3.2 What is NOT code inspection?

Testing and code inspection are complementary. Looking at a value/input cell is an indication of testing and not code inspection. Also when a formula is posted and then edited next, we avoid calling it code inspection since it could imply bug fixing rather than code inspection.

3.3 Motivation to Investigate Code Inspection

In the experiment conducted during summer 2006, we gave the participants a post-session questionnaire which had an open-ended question about what strategies they said they followed in finding and fixing errors. We found that more females than males mentioned code inspection as one of their strategies, but the result was not statistically significant. In the first attempt at analysis, we then decided to look for non-code inspection, i.e., editing input cells in order to develop test cases for testing. We supposed that participants who occupied themselves more in testing would do less code inspection. We found that, indeed, males edited input cells significantly more often than females.

In an independent study conducted by Fern et al. using data-mining as a tool to find different debugging behavior patterns, it was found that females used code inspection significantly more often than males did [Fern et al.]. Hence, the above two instances convinced us that investigating code inspection further would not be fruitless.

3.4 Qualitative Analysis

There was a set of participants who had mentioned code inspection as their strategy in the questionnaire. What about the rest of the participants? Could we say with certainty that the rest did not use code inspection as their strategy? The answer is NO. It is quite possible that these participants were using the strategy without actually mentioning it.

Forms/3 has a provision to replay the transcripts using a tool called the script-player. This java-based tool enabled us to view video of an entire debugging session of each participant of the set X. While replaying the transcripts we looked for evidence of code inspection.

To do this, we developed criteria to which we strictly adhered in order to determine whether the strategy of code inspection was employed by the participants. We used two measures: total number of code inspection instances and total number of formulas displayed.

We defined an “instance of code inspection” to be in progress if:

1. 2 or more formulas were displayed simultaneously without editing on two or more of the displayed ones.

Or

2. A formula was displayed, then “undisplayed” without editing and this behavior continued consecutively for more than 2 formulas with no intervening actions.

An instance was defined as ending when the formulas were “undisplayed” or some formula was edited.

We also counted the total number of formulas displayed for each session. In order to ensure that we did not mistake testing behavior as code inspection, we cross-checked by ensuring that it was formulas and not values that were displayed.

3.5 From Qualitative To Quantitative

By replaying the transcripts, we counted the number of code inspection instances and number of total displayed formulas during the session of the Gradebook problem for successful females and successful males using the above rules. We categorized the participants as “successful” or “unsuccessful”, depending on whether they fixed the median number of bugs (5.5). Because the qualitative mechanism was time-consuming, we restricted our replays to the 30 successful participants.

Since the formulas of the Gradebook spreadsheet were spatially more spread out that that of the Payroll spreadsheet, it enabled us to observe the code inspection instances and the displayed formulas more easily in the former. Hence we chose the Gradebook spreadsheet for the qualitative observation. We then determined statistically using ANOVA whether there was any difference between the successful females and successful males in these measures.

3.6 Results

Groups	Number of Code inspection instances	Total displayed formulas
16 Successful Males, 14 Successful females	SF>SM SM = 6.5294(1.99) SF = 7.8571(2.24) F(1,28) = 4.99 **p = 0.03350397**	SF>SM SM = 14.5(5.08) SF = 19(5.37) F(1,28) = 5.541 **p = 0.02582071**

Table 1. Numbers of code inspection instances and total displayed formulas (F- Females, M-Males, S-Successful, and U- Unsuccessful). (ANOVA)

As Table 1 shows, by either measure, successful females significantly used code inspection more often than successful males.

3.7 Discussion

Why did successful females use more code inspection than successful males?

One of the reasons for the above could be the comprehensive processing that females tend to adopt in problem solving, as proposed by the selectivity hypothesis [Meyers-Levy 1989]. In contrast to males, females tend to maximize the comprehensiveness of their information processing, looking for multiple, subtle cues, paying attention to detail, and making elaborative inferences. The hypothesis predicts that females are likely to employ detailed, elaborative information processing strategies in both simple and complex decision tasks. Males on the other hand, have a tendency to use simple heuristics in information processing (e.g., single cues that are readily available) in order to reduce cognitive load.

Code inspection would provide an opportunity to understand the big picture before proceeding into smaller sub tasks. This could be one of the reasons why females use the strategy of code inspection. Hence providing software which supports code inspection could help females perform better in the task of debugging spreadsheets and other such problem solving domains.

4. What Were Successful Males Doing? Testing

4.1 What is testing?

"Testing" is defined in the software engineering field as executing a program with different inputs, with the specific intent of finding errors [Pressman et al 2005.]. In spreadsheets, the program is the collection of formulas and the errors are the bugs in the formulas. Hence, testing in this environment can be defined as executing the spreadsheet with the intent of finding bugs, by modifying the values of the spreadsheet to see the answers produced.

In the summer 2006 experiment, a final post-session questionnaire included an open-ended question that asked the participants what they perceived their own strategies to be for finding and fixing errors. Many participants described their strategy in terms of testing, or said that a testing-oriented feature or example values handout was important to their strategy. No significant difference was found in what the males versus females said about this. However, there was a possibility that some participants used testing as their strategy but did not mention it in the questionnaire. In order to consider this, we decided to look at what they actually did.

4.2 How to measure testing?

4.2.1 Percent testedness:

The debugging features that were present in this experiment were part of WYSIWYT ("What You See Is What You Test"). WYSIWYT is a collection of testing and debugging features that allow users to incrementally "check off" or "X out" values that are correct or incorrect, respectively [Burnett et al. 2004]. Whenever users decide a cell's value is correct, they can place a checkmark (√) in the decision box at the corner of the cell they observe to be correct: this communicates a successful test. Behind the scenes, checkmarks increase the "testedness" of a cell according to a test adequacy criterion based on formula expression coverage (described in [Rothermel et al. 2001]), and this is depicted by the cell's border becoming more blue.

Figure 4 shows an example of the progress bar. The progress bar reflects the testedness of the entire spreadsheet at a particular instance. We used the total percent testedness as one measure of testing activity.

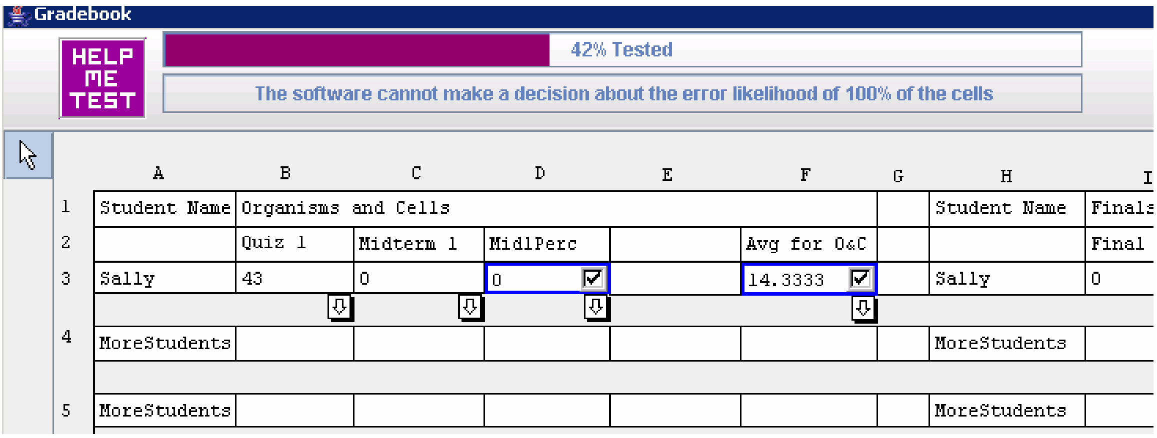


Figure 4. Progress bar in the Forms/3 environment.

4.2.2 Testing-related user actions

Value manual editing - The participant can change the input values in the value cells (Figure 5). One source of ideas for possible input values was the example values handout , which showed two sets of sample inputs and resulting values for every cell.

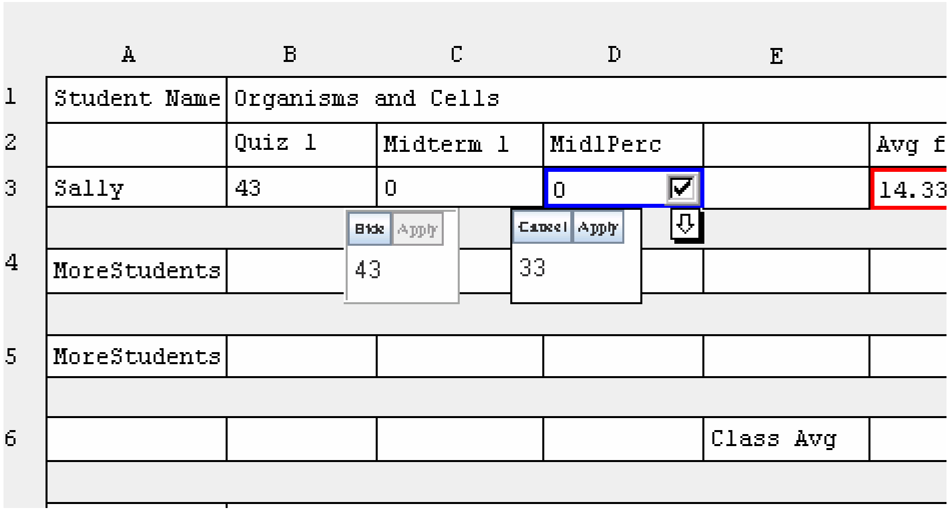


Figure 5. Value editing in the Forms/3 environment

Help Me Test -

The "Help Me Test" (HMT) feature [Fisher II et al. 2002, Wilson et al. 2003] is provided to help users find additional test cases. Sometimes it can be difficult to find test values that will cover the untested logic in a collection of related formulas. If the user pushes the Help Me Test button (upper left of Figure 4), the system tries to find inputs that will lead to coverage of untested logic in the spreadsheet, about which users can then make testing decisions. Help Me Test is not fully automated testing but rather scaffolding: it provides new test inputs, but does not make decisions about the outputs that result, so does not actually "test" the spreadsheet. Although we cannot identify testing behavior using HMT clicks alone, they can be used along with value edits to cross-check the amount of a user's testing behavior.

4.3 Results

4.3.1 Gradebook testing

Focusing first on the Gradebook problem, males manually edited value cells significantly more often than females in the Gradebook problem (Table 2). Also, the successful males' total percent testedness for the Gradebook problem was significantly higher than that of the successful females. Finally, the successful males had significantly higher testedness than unsuccessful males. In short, for males, testing was key to their success, but it was not implicated in the females' success at all.

Group	**Total value edits**	**Total Percent-testedness**
24 Males , 37 Females	M>F M 24.12(15.53) F 17.10(10.18) F(1,59)=4.556 **p= 0.03695113**	M>F M 0.756(0.1910) F 0.651 (0.2175) F(1,59)=3.75 p = 0.05873
14 Successful Females, 23 Unsuccessful Females	SF>UF SF 19.62(11.16) UF 15.19(8.95) F(1,35)=1.75 p = 0.4419845	SF>UF SF 0.652(0.2822) UF 0.651(0.1677) F(1,35)=0.0001 p= 0.9900
16 Successful Males, 8 Unsuccessful Males	SM>UM SM 28.93(15.59) UM 16.11(12.35) F(1,22)=4.40 p = 0.05993785	SM>UM SM 0.825 (0.1613) UM 0.642(0.2047) F(1,22)=6.31 **p= 0.0197**
16 Successful Males, 14 Successful Females	SM>SF SM 28.93(15.59) SF 19.62(11.16) F(1,29)=3.68 p = 0.07557929	SM>SF SM 0.825 (0.1613) SF 0.652 (0.2822) F(1,29)=6.82 **p= 0.0141**
8 Unsuccessful Males, 23 Unsuccessful Females	UM>UF UM 16.11(12.35) UF 15.19(8.95) F(1,28)=0.05 p=0.934031	UF>UM UM 0.651(0.2047) UF 0.642(0.1677) F(1,28)=0.009 p= 0.9219

Table 2. The p values of the different groups in Gradebook (F- Females, M-Males, S-Successful, and U- Unsuccessful). (ANOVA)

4.3.2 Payroll testing

The Payroll problem had results similar to those of Gradebook. Males manually edited value cells significantly more often than females. Also, successful males manually edited value cells significantly more often than successful females (Table 3).

Group	Total Value edits	Total Percent-testedness
24 Males , 37 Females	M>F M 24.54(8.78) F 14.54(18.76) F(1,59)=7.89 **p = 0.00670956**	M>F M 0.736(0.1843) F 0.681(0.1318) F(1,59)=1.77 p = 0.1879
14 Successful Females, 23 Unsuccessful Females	SF>UF SF 17.00(9.06) UF 12.87(8.38) F(1,35)=2.03 p = 0.1627338	SF>UF SF 0.699(0.1341) UF 0.669(0.1326) F(1,35)=0.441 p = 0.5106
16 Successful Males, 8 Unsuccessful Males	SM>UM SM 29.79(21.14) UM 17.20 (12.29) F(1,22)=2.83 p = 0.1064967	SM>UM SM 0.786 (0.1725) UM 0.666(0.1886) F(1,22)=2.63 p = 0.1188
16 Successful Males, 14 Successful Females	SM>SF SM 29.79(21.14) SF 17.00(9.06) F(1,27)=4.58 **p = 0.04134435**	SM>SF SM 0.786 (0.1725) SF 0.699 (1.1342) F(1,27)=2.94 p = 0.0973
8 Unsuccessful Males, 23 Unsuccessful Females	UM>UF UM 17.20(12.29) UF 12.87(8.38) F(1,30)=1.36 p = 0.1784454	UF>UM UM 0 .669(0.1886) UF 0.666(0.1326) F(1,30)=0.002 p = 0.9619

Table 3. The p values of the different groups in Payroll (F- Females, M-Males, S-Successful, and U- Unsuccessful). (ANOVA)

4.3.3 Total testing

Males did significantly more edits to value cells than females. Also, successful males did significantly more edits to value cells than successful females (Table 4).

We cross-verified this by combining the HMT clicks with the value edits. Here again, males did significantly more combined Help Me Test clicks and value edits than females. Also, successful males did this significantly more often than successful females. This indicated the robustness of the result: it does not matter exactly how we define testing activity, the results come out the same.

Group	**Total value edits**	**Manual value edits +HMT**	**Total Percent-testedness**
24 Males , 37 Females	M>F M 48.67(33.02) F 31.65(15.52) F(1,59)=7.61 **p = 0.007706**	M>F M 53.2(31.60) F 36.21(16.34) F(1,59)=7.60 **p = 0.007718**	M>F M 0.746(0.184) F 0.666(0.131) F(1,59)=3.47 **p = 0.0415**
14 Successful Females, 23 Unsuccessful Females	SF>UF SF 36.36(19.34) UF 28.78(12.28) F(1,35)=2.13 p = 0.1527884	SF>UF SF 39.21(19.16) UF 34.39(14.52) F(1,35)=0.75 p = 0.3916012	SF>UF SF 0.662(0.1820) UF0.669(0.0940) F(1,35)=0.22 p = 0.8821
16 Successful Males, 8 Unsuccessful Males	SM>UM SM 59.68 (31.99) UM 26.62(22.93) F(1,22)=7.06 **p = 0.0143885**	SM>UM SM 64.37 (29.85) UM 30.87(22.72) F(1,22)=7.75 **p = 0.0106**	SM>UM SM 0.796(0.141) UM0.669(0.173) F(1,22)=5.06 **p = 0.0347**
16 Successful Males, 14 Successful Females	SM>SF SM 59.68(31.99) SF 36.36(19.34) F(1,28)=5.88 **p = 0.021984**	SM>SF SM 64.37 (29.85) SF 39.21 (19.16) F(1,28)=7.29 **p = 0.01160**	SM>SF SM 0.796(0.141) SF 0.662(0.182) F(1,28)=5.08 **p = 0.0321**
8 Unsuccessful Males, 23 Unsuccessful Females	UF>UM UM 26.62(22.93) UF 28.78(12.28) F(1,29)=0.114 p = 0.7375781	UF>UM UM 34.39(22.72) UF 30.87(14.52) F(1,29)=0.25 p = 0.6154	UF>UM UM 0.669(0.147) UF 0.666(0.173) F(1,29)=0.205 p = 0.6533

Table 4. The p values of the different groups with both Gradebook and Payroll combined (F- Females, M-Males, S-Successful, and U- Unsuccessful). (ANOVA)

4.3.4 Is testing behavior a predictor of the total bugs fixed?

As a cross-check for our statistics on the successful/unsuccessful groups, we ran Linear Regression tests on the following:

1. Percent testedness as a predictor of bugs fixed

Group	**Gradebook**	**Payroll**	**Total**
Males	R-Squared: 0.2281 F(1,22)=6.5 Beta=0.0477 **p =0.01827**	R-Squared: 0.1035 F(1,22)=2.541 Beta=0.0445 P=0.1252	R-Squared: 0.2127 F(1,22)=5.94 Beta=0.0311 **p= 0.0239**
Females	R-Squared:0.003805 F(1,35)=0.1337 Beta=-0.0075 p=0.7169	R-Squared:0.00999 F(1,35)=0.3532 Beta= 0.0091 P=0.5561	R-Squared: 0.005067 F(1,35)=0.1783 Beta=-0.0033 p=0.6755

Table 5. Percent testedness as a predictor of the bugs fixed. (Linear regression)

As Table 5 indicates, Percent testedness was a predictor of the total bugs fixed for males in the Gradebook problem and in total. On the other hand this was not a predictor for females.

We ran multiple regression tests on gender, percent testedness and the interaction between the two, being predictors of the bugs fixed. Although we did not get any significance in Payroll, in Gradebook the slopes of this interaction for males and females differed significantly (multiple regression: $F(3,57)=7.009$, $p<0.0292$, $\beta=2.6412$, $R^2=0.2695$) and the slopes were marginally different (multiple regression: $F(3,57)=5.722$, $p<0.0805$, $\beta=4.1882$, $R^2=0.2315$) in total. This indicates that the testing strategy helped males but not females in debugging the Gradebook spreadsheet (illustrated in the Figure 7).

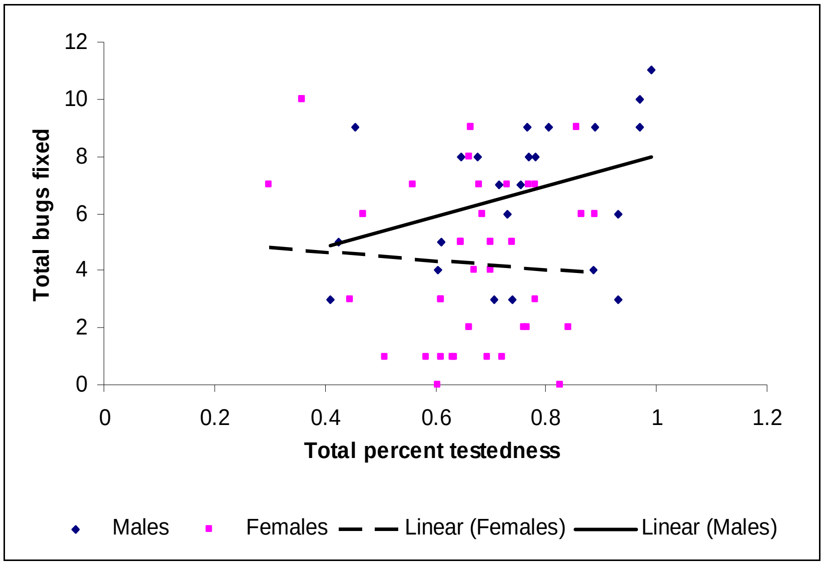


Figure 6. Linear regression of percent testedness as a predictor of bugs fixed in total.

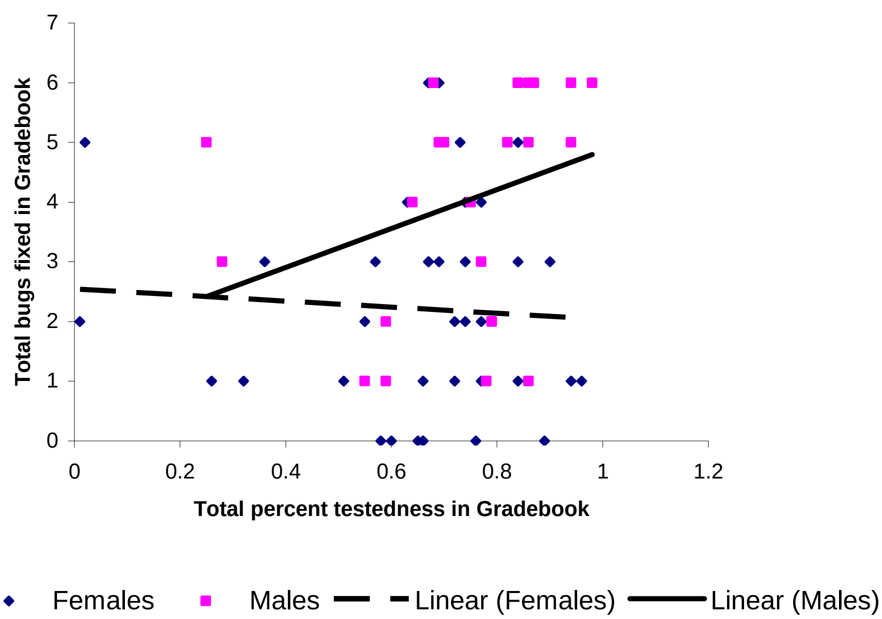


Figure 7. Linear regression of percent testedness and bugs fixed in Gradebook.

2. Number of value edits as a predictor of bugs fixed

Group	Gradebook	Payroll	Total
Males	R-Squared: 0.1658 F(1,22)=4.37 Beta=0.0501 **p=0.0482**	R-Squared: 0.08186 F(1,22)=1.961 Beta=0.0203 p=0.1753	R-Squared: 0.2095 F(1,22)=5.82 Beta=0.0348 **p=0.0243**
Females	R-Squared: 0.006374 F(1,35)=0.225 Beta=0.0140 p=0.6385	R-Squared: 0.03485 F(1,35)=1.264 Beta=0.0315 p=0.268	R-Squared: 0.02721 F(1,35)=0.978 Beta=0.9043 p=0.3293

Table 6. Value edits as a predictor of the bugs fixed. (Linear regression)

As Table 6 shows, the number of value edits was a predictor of the bugs fixed for males in the Gradebook problem and in total. This is also illustrated in the Figure 8. However, this was not a predictor for females. Multiple regression tests, however, did not show significance on gender, total value edits or the interaction between the two, being predictors of the bugs fixed. Thus, from the linear regression statistics, we can conclude only that this debugging strategy testing predicted the success for males.

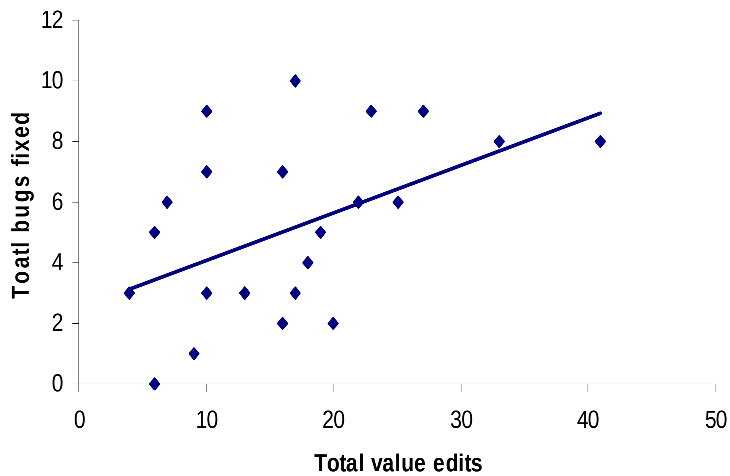


Figure 8. Linear regression of total value edits and bugs fixed for males

4.4 Discussion

Why did successful males do more testing than successful females?

There could be more than one explanation for the above question. Code inspection and testing are two approaches to find bugs in the spreadsheet. Since successful females did significantly more code-inspection successful males, it is possible that this could be the reason for the successful females to do less testing.

A rival explanation would be such that since females tend to employ the comprehensive strategy [Meyers-Levy 1989] towards problem solving, it is possible that they initially tried to understand the code using the code inspection strategy and then migrated to testing. Males on the other hand, tend to process information selectively which would enable them to start testing the spreadsheet much earlier than females during the debugging task. It would be interesting to analyze the effect of time on the testing behavior as part of future work.

It is also interesting that the difference between males' and females' strategies were more pronounced in Gradebook than Payroll. It is possible that Payroll's logic was less conducive to understanding it by reading the formulas, forcing females toward more reliance on testing despite their inclination toward code inspection. This is another question that would be interesting to pursue in future work.

5. More on Successful Males and Females

After having observed the testing and code-inspection behavior with respect to the debugging success of males and females, we were curious to know if a strategy usage was influenced by variables other than gender. Also, we knew that testing and code-inspection was widely used by males and females respectively. However, we wanted to know how the two strategies interacted with each other. Hence we decided to analyze the:

1. Influence of background variables on strategy usage

2. Ties of one strategy to the debugging success in the presence of the other strategy as well as the interaction of the two strategies.

We decided to restrict the analysis to successful participants only, since we were primarily interested in the behavior patterns of end users who were successful in their debugging. Moreover, in order to do a comparative analysis of both strategies, code-inspection and testing, we needed to have a uniform set of population for both testers and code-inspectors. Since we had earlier analyzed the successful participants alone for code-inspection behavior, we decided to keep the unit of analysis as the successful males and females.

There were three related background variables namely pre-self-efficacy, experience and GPA. We ran linear regression tests on various success groups of both genders. We found that one measure of the code-inspection strategy (total formulas opened) was negatively predicted by pre-self-efficacy for successful males (linear regression: $F(1,14)=5.03$, $p<0.0416$, $\beta=0.7073$, $R^2=0.26$). The other measure of the code-inspection strategy (count of instances) was marginally negatively predicted by pre-self-efficacy for successful males (linear regression: $F(1,14)=4.065$, $p<0.063$, $\beta=-0.256$, $R^2=0.225$). The same was marginally positively predicted for successful females (linear regression: $F(1,12)=3.28$, $p<0.074$, $\beta=0.2041$, $R^2=0.2418$). No other background variables had any effect on the strategy usage for both groups. This further strengthened the effect of the “gender” variable on the usage of strategies.

The second part of our analysis involved investigating whether the strategies and their interaction influenced the debugging success for the successful participants. We found that the code-inspection strategy (count instances) negatively predicted total bugs fixed for successful females (linear regression: $F(1,12)=5.25$, $p<0.0407$, $\beta=-0.031$, $R^2=0.30$. However, previously, we had found that significantly more number of successful females mentioned code-inspection as their strategy than unsuccessful females (Fisher’s Exact: $p<0.03$; 13/14 successful females and 13/23 unsuccessful females). These two results seemed to indicate different things about code-inspection. Similarly, we had found that the testing strategy predicted the debugging success for males (linear regression: $F(1,22)=5.94$, $p<0.0231$, $\beta=0.0311$, $R^2=0.2127$) . But when we ran the same tests on its sub-group (successful males), we did not get significance. Hence, although the testing and the code-inspection strategy helped males and females to be successful respectively, there is no evidence that its *extent* directly influenced the extent to which they were successful.

Multiple regression on testing, code-inspection and the interaction between the two predicting the total bugs fixed, produced no main effects, but a significant interaction effect for successful males (Multiple regression: $F(3,12)=1.813$, $p<0.0497$, □=0.0110, R□=0.3118). This says that the effects of the testing strategy on total bugs fixed depended on in the effects of the code inspection strategy and vice versa, for successful males. No significant effects were found for successful females. This result is interesting since it talks about the interaction of the strategies. It however does not say anything about the effect of the combination of strategies on debugging success. Hence, it would be interesting to analyze the effect of the combination of the strategies as part of future work.

6. What Were the Unsuccessful People doing?

6.1 What is "fixing formulas"?

"Fixing formulas" is editing formulas that have errors so as to correct them. There are two kinds of formulas edits that might be made: "buggy" and "inappropriate". The buggy edits are changes to formulas that contain errors, while the inappropriate edits are the changes to formulas with no errors.

6.2 Fixing formulas as a strategy

In the open-ended questionnaire of the summer 2006 experiment, some participants mentioned "fixing formulas" as their strategy. When we looked closer, we saw that most of the participants who mentioned this strategy were unsuccessful females. This made us wonder whether this strategy interfered with their debugging task.

When we ran ANOVA on the total number of edits to buggy formulas between successful/unsuccessful groups of females and males, we did not get any significant difference between the genders. This raised the following questions: Were the females who mentioned fixing formulas as their strategy actually editing the buggy formulas or were they mistakenly spending their time on inappropriate edits and thereby introducing new bugs?

We then looked at the inappropriate edits by males and females. Although there was no significant difference in the Gradebook problem, in the Payroll problem females did inappropriate edits significantly more often than males (males: 1.625(2.20) females: 3.486(3.11); ANOVA: $F(1,59)=6.45$, $p<0.0136$). The total was marginally significant that females did inappropriate edits more often than males in total (males: 5.833(5.04) females: 7.919(3.67); ANOVA: $F(1,59)=3.04$, $p<0.0860$).

Linear regression tests showed that mentioning the fixing formula strategy was predictive of inappropriate edits for females (linear regression: $F(1,35)=2.5803$, $p<0.0214$, $\beta=4.22$, $R^2=0.14$). On the other hand, this did not have any predictive value for males (linear regression: $F(1,22)=0.3834$, $p<0.5421$, $\beta=1.00$, $R^2=0.017$). Hence, when a female mentioned fixing formulas as her strategy, there was a high chance of her editing inappropriately. Since many of the females (8/10) who mentioned fixing formula as a strategy were unsuccessful, it was possible that these unsuccessful females spent their time editing the wrong formulas, thereby introducing new bugs. One the other hand, only 3 out of 8 males who mentioned this strategy were unsuccessful. These results are consistent with a previous study that indeed found that females introduced significantly more bugs than males did [Beckwith et al. 2005], although they fixed the same number of bugs as males.

6.3 Discussion

We have looked into strategies like testing (helping males) and code-inspection (helping females). Looking at strategies that were tied negatively to the success of females, it is clear that one such strategy was "fixing formulas".

Is "fixing formulas" really a strategy? If it is not, the implication is that people who mentioned this simply spent their time in an ad hoc manner, editing formulas without being able to differentiate formulas from others. This may be a consequence of them lacking strategies like code-inspection and testing which helped others identify the buggy formulas. If it does count as a strategy, it was indeed a poor one. Whether are not it was a strategy, it was a pitfall to which females were particularly susceptible.

Males on the other hand, were not particularly susceptible to this pitfall, but they have a different one. Instead previous studies have shown that males tinkered more than females and that, unlike females tinkering, males' tinkering was often counterproductive to their effectiveness in debugging [Beckwith et al. 2006]. Taken together, these results may suggest that pitfalls for male and female end-user programmers are different.

7. Conclusion

This thesis described our investigation on the strategies used by end-user programmers while debugging. We found that there were significant gender differences in the strategies males and females were using. Further, the debugging strategies that worked well for the males were not the same ones that worked well for the females.

On analyzing two major strategies that the end-user programmers talked about we have the following interesting results:

Testing: Quantitative analysis of the behavior logs of participants showed that this strategy was used widely by successful males. The usage of this strategy helped in the debugging success of males. On the other hand, this strategy did not have any significant impact on the success of the females.

Code Inspection: By qualitative analysis of the scripts, it was found that the code inspection strategy was preferred mainly by successful females. However successful males preferred it less compared to the successful females

Hence, it was revealed that females mostly inspected the code for finding errors while males preferred to test for the same.

Besides these major strategies, we also explored a minor strategy “fixing formulas”. We found that the females who were supposedly “fixing formulas” ended up spending their time editing inappropriate formulas and hence possibly being unsuccessful.

In end-user programming environments, we noticed that although the testing strategy is well supported, the code-inspection strategy is not. Hence, these results show that end-user programming environments have the need for improving their support for debugging strategies especially the ones used by the female end-user programmers.

Bibliography

[Bandura 1977] Bandura, A. Self-efficacy: Toward a unifying theory of behavioral change. *Psychological Review 8*(2), 1977, 191-215.

[Beckwith et al. 2005] Beckwith, L., Burnett, M., Wiedenbeck, S., Cook, C., Sorte, S. and Hastings, M. Effectiveness of end-user debugging software features: Are there gender issues? *In Proc. of ACM Conference on Human Factors in Computing Systems (CHI'05)*. 2005, 869-878.

[Beckwith et al. 2006] Beckwith, L., Kissinger, C., Burnett, M., Wiedenbeck, S., Lawrance, J., Blackwell, A. and Cook, C. Tinkering and gender in end-user programmers' debugging. *In Proc. of ACM Conference on Human-Computer Interaction (CHI'06)*. 2006, 231-240.

[Blackwell 2002] Blackwell, A. First steps in programming: a rationale for attention investment models. In *Proc. IEEE Human-Centric Computing Languages and Environments* (2002), 2-10.

[Burnett et al. 2004] Burnett, M., Cook, C. and Rothermel G. End-user software engineering. *Comm. ACM 47*, 9 (2004), 53-58.

[Busch et al. 1995] Busch, T. Gender differences in self-efficacy and attitudes toward computers. *J. Educational Computing Research 12* (1995), 147-158.

[Compeau et al. 1995] Compeau, D. and Higgins, C. Computer self-efficacy: Development of a measure and initial test. *MIS Quarterly 19*, 2 (1995), 189-211.

[Czerwinski et al. 2002] Czerwinski, M., Tan, D. and Robertson, G. G. Women take a wider view. *In Proc. of ACM Conference on Human-Computer Interaction (CHI'02),* 2002, 195-202.

[Fennema et al. 1998] Fennema, E., Carpenter, T., Jacobs, V., Franke, M., Levi, L., A Longitudinal Study of Gender Differences in Young Children’s Mathematical Thinking, *Educational Researcher*, 27(5), 1998, 6-11.

[Fern et al. 2007] Fern, X., Komireddy, C., and Burnett, M. Mining interpretable human strategies: A case study, In *Proc. Int. Conf. Data Mining*, IEEE (2007), to appear.

[Fisher II et al. 2002] Fisher II, M., Cao, M., Rothermel, G., Cook, C. and Burnett, M. Automated test case generation for spreadsheets. *In Proc. of Int'l. Conf. on Software Engineering*. 2002, 141-151.

[Gallagher et al. 2000] Gallagher A., De Lisi R., Holst P., McGillicuddy-De Lisi A., Morely M., Cahalan C. Gender differences in advanced mathematical problem solving, *Journal of Experimental Child Psychology* 75(3), 2000, 165-190.

[Grigoreanu et al. 2006] Grigoreanu, V., Beckwith, L., Fern, X., Yang, S., Komireddy, C., Narayanan, V., Cook, C., and Burnett, M. Gender differences in end-user debugging, revististed: What the miners found. In *Proc. Visual Languages and Human-Centric Computing*, IEEE (2006), 19-26.

[Halpern et al. 2000] Halpern, D. F. *Sex Differences in Cognitive Abilities*, 3rd Edition. Lawrence Erlbaum Associates, Inc., NJ, 2000.

[Huff 2002] Huff, C., Gender, software design, and occupational equity, *ACM SIGCSE Bulletin*, 34 (2), 2002, 112-115.

[Kimura et al. 1999] Kimura, D. *Sex and Cognition*. MIT Press, Cambridge, Mass, 1999, pp. 1-66.

[Meyers-Levy 1989] Meyers-Levy, J. Gender differences in information processing: A selectivity interpretation. In Cafferata, P. and Tybout, A. (Eds.) *Cognitive and Affective Responses to Advertising*. Lexington Books, Lexington, MA, 1989, 219-260.

[Panko 1998] Panko, R. What we know about spreadsheet errors. *J. End User Computing 10*, 2 (1998), 15-21.

[Pressman et al. 2005] Pressman, R. *Software Engineering (3rd edition)Practitioner's Approach*, McGraw-Hill, Inc., New York, NY, 1992.

[Prabhakararao et al. 2004] Prabhakararao, S., Cook, C., Ruthruff, J., Creswick, E., Main, M., Durham, M., and Burnett, M. Strategies and behaviors of end-user programmers with interactive fault localization. In *Proc. Visual Languages and Human-Centric Computing*, IEEE, 2004, 15-22.

[Rothermel et al. 2001] Rothermel, G., Burnett, M., Li, L., DuPuis, C. and Sheretov, A. A methodology for testing spreadsheets. *ACM Trans. Software Engineering and Methodology* 10(1), 2001, 110-147.

[Sandstrom et al. 1998] Sandstrom, N. J., Kaufman, J., & Huettel, S. A. Males and females use different distal cues in a virtual environment navigation task. *Cognitive Brain Research*, 6, 1998, 351-60.

[Waller et al. 1998] Waller, D., Hunt, E., and Knapp, D. The transfer of spatial knowledge in virtual environment training.Presence: *Teleoperators and Virtual Environments*, 7(2), 1998, pp. 129-14.

[Whitworth et al. 2002] Whitworth, J.E., Price, B.A. and Randall, C.H. Factors that affect college of business student opinion of teaching and learning. *J. Education for Business 77*, 5 (2002), 282-289.

[Wilson et al. 2003] Wilson, A., Burnett, M., Beckwith, L., Granatir, O., Casburn, L., Cook, C., Durham, M. and Rothermel, G. Harnessing curiosity to increase correctness in end-user programming. *In Proc. of ACM Conference on Human Factors in Computing Systems.* 2003, 305–312.

Appendix A: Tutorial Materials

Introduction

Hi, my name is __________________, and I will be leading you through today's study.

The other people involved in this study are Dr. Margaret Burnett, Dr. Susan Wiedenbeck, Valentina Grigoreanu, Laura Beckwith, Neeraja Subrahmaniyan, and Karin Bucht. .

Before we start, I'd like to remind you to please turn off your cell phones.

Just so you know, I will be reading through this script so that I'm consistent in the information I provide you and the other people taking part in this study, for scientific purposes.

The aim of our research is to help people create correct spreadsheets. Past studies indicate that spreadsheets can contain several errors like incorrectly entered values and formulas. Our research is aimed at helping users find and correct these errors.

For today's experiment, I'll lead you through a brief tutorial of Forms/3, our research spreadsheet software, and then you will have a few experimental tasks to work on.

(Do this next paragraph only on a "just in case" basis – if most have completed everything by the time the tutorial begins, don't let them become bored!) But first, if you haven't already done so, please read the text of the "Informed Consent Form" that you currently have in front of you. You only need to sign one of the two copies; the other is for you to keep.
(Give them time to read the form and sign it).

Please do NOT discuss this study with anyone. We are doing later sessions and would prefer the students coming in not to have any advance knowledge.

Questions?
If you have any questions, contact Dr. Margaret Burnett, who's name and contact information is on the paperwork you will take away with you.

Background Questionnaire *(have them fill it out, collect, check)*

Tutorial

In this experiment, you will be working with the spreadsheet language Forms/3. To get you familiarized with its features, we're going to start with a short tutorial. After the tutorial, you will be given two different spreadsheets and will be asked to test those spreadsheets and, if you find any errors, fix them.

As we go through this tutorial, I want you to ACTUALLY PERFORM the steps I'm describing. Please pay attention to your screen while you do the steps.
Please do not get ahead of the tutorial, you might miss important points.
If you have any questions, please raise your hand. *(Point to the driver)* will also be performing the steps on the overhead, as we go through the tutorial.

We are going to teach you several features that will help you with finding and fixing errors. So let's begin our tour of Forms/3's features.

Here's some information about the first spreadsheet.
(Hand out PurchaseBudget Description - First the Description, then the Samples.)

You have two handouts in front of you. One is a description of how this spreadsheet should behave. The other handout provides example correct values. Read through both of these handouts about the "PurchaseBudget" spreadsheet now.

(Wait for them to read – read yourself silently to not cut them way short)

Now open the PurchaseBudget spreadsheet by clicking on the bar labeled PurchaseBudget at the bottom of the screen.

This is a Forms/3 spreadsheet. There are a few ways in which Forms/3 spreadsheets look different than the spreadsheets you may be familiar with. Most notably, you can see that some cells have colored borders.

Let's find out what the red color around the borders means. Rest your mouse on top of the border of cell D…4 *(wave the mouse around the cell and then rest mouse on border).* A tooltip will pop up and tell us what this color means. *(No Pause)* If you had trouble getting it to come up, move your mouse out of the cell and come back over onto the border a bit faster. Can anyone tell me what the tooltip says? *(PAUSE, look for a hand.)* Yes, it means that the cell has not been tested.

You might be wondering what testing has to do with spreadsheets. Well, it is possible for errors to exist in spreadsheets, but what usually happens is that they tend to go unnoticed.

So, the red border around the cells is just telling us that the cell has not been tested. Testing is trying out different values, leading to different situations, to see if the answers are coming out right.

Observe that both cell C2 and cell C3 have black borders *(wave mouse around cells)*. These cells with black borders only contain values; they do not have formulas, so they can't be tested. Cells with formulas have colored borders.

One of the sheets you have in front of you is titled "escription for the PurchaseBudget Spreadsheet Problem". *(Wave it around and show it to them)* The first sentence of the callout that points to cell D4 says that "The Cost Comb. is the combined cost of pens and paper." *(Point at it.)* Since we have 0 pens and 0 reams of papers, and 0+0 is 0, we decide that the value of "0" in cell D4 is correct. Set your mouse over the small box with a question mark in the upper-right-hand corner of cell D4. Can anyone tell me what the tooltip says? *(PAUSE, wait for answer.)* Yes, it says to click if we decide whether the value is correct or wrong. It also tells us that these decisions help test and find errors.

So let's click the question mark in this decision box. *(mouse clicks)* Now a line of four options has popped up. What do these four choices mean? You would choose the right-most checkmark, *(point to with mouse)* if you were sure the cell's value is correct. You would choose the left-most x-mark (*point to with mouse)* if you decided that the value is definitely wrong. The inner checkmark is for cases when you think a value *might* be right, but are not sure. And the inner x-mark is for cases where you think a value might be wrong, but you are not sure. We'll place an x-mark toward the end of the tutorial.

The decision that we're going to make now might not be the correct one – you will have to decide whether this decision is correct when you will be working on your own, later on in the session. But, for the purposes of teaching you features in this tutorial, let's say that you are sure that the value is correct. Click on the right-most checkmark. (*mouse clicks)* Notice what happened. Three things changed. A checkmark replaced the question mark in the decision box *(wave mouse)*. The border colors of some cells changed—some cell borders turned blue (*point to borders)*, and the spreadsheet percent testedness at the top of your screen increased to 22% *(point to it)*.

Sometimes you may need to remove a checkmark, for example, if you accidentally place a checkmark in the decision box and the value was really wrong, or if you haven't seen the changes that occurred. To "uncheck" the decision, click on that checkmark. *(Pause)* Everything goes back to how it was. The cells' borders turned back to red, the percent testedness bar dropped back to 0% and a question mark reappeared in the decision box.

Put the checkmark back in the decision box for D4. (*Wait for mouse clicks)*.

You may have noticed that the border colors of cells D2 and D3 are both blue. Now let's find out what the blue border indicates by holding the mouse over cell D2's border in the same way as before. *(PAUSE)* The message tells us that the cell is fully tested. Also notice the blank decision box in cells D2 and D3. What does that mean? Position your mouse on top of the box to find out why it is blank. A tooltip pops up that says we have already made a decision about this cell. *(Pause – Act surprised!)* How did this happen? We haven't made a decision about D2 and D3 yet! Let's find out.

Position your mouse cursor in the middle of cell D4 and click the scroll wheel (show it). (*Wait for mouse clicks)* Colored arrows appear. To better see all of the arrows, it sometimes helps to open the cell's formula. To see D4's formula, move your mouse to the arrow right underneath the checkmark in cell D4. It says "Click here to show formula." Click on this tab at the bottom right of the cell. Its formula opened up.

Click the scroll wheel again on any one of these arrows *(PAUSE)*—it disappears. Now, click the scroll wheel again on cell D4 —all the other arrows disappear. (*mouse clicks)* Now bring the arrows back again by re-clicking the scroll wheel on D4. *(Wait for mouse clicks)*

Notice the arrows are colored the same way as the cell borders. Move your mouse over to the topmost blue arrow and hold it there until a tooltip appears. *(pause)* It first explains that the arrow is showing a relationship that exists between D2 and D4. The value in D2 goes into or contributes to the answer for D4. *(PAUSE)* The tooltip also explains that the relationship between cell D2 and cell D4 is fully tested. *(ANOTHER PAUSE)* If you can't tell what cells an arrow is pointing to and from, the tooltip will tell you what those cells are.

This explains why if you mark one cell value as being correct, as you did with D4, and there are other cells contributing to it, such as D2 and D3, those cells' borders will also be colored as tested. *(PAUSE)*

We don't need those arrows on D4 anymore, so let's hide them by clicking the scroll wheel on cell D4. Let's also hide the formula, since we don't need it anymore. To hide the formula, hit the "Hide" button above it. (*PAUSE)*

We are now going to change some values and formulas. First, open the formula for cell B2, the number of pens on hand. (*WAIT FOR MOUSE CLICKS.)* This cell's formula is just a value, and we'd like to try a different value. We've got an example correct spreadsheet here with some values. *(Wave around.)* It says "25" for cell B2. *(Point to it.)* So, let's use this value, since it's an example we can refer to. Try changing the value to 25 and click the "Apply" button. (*PAUSE)*

Cell E…2 tells us whether we have enough pens. We need more than 68 boxes of pens. (*PAUSE)* So the answer "pen quantity ok" is not right. Don't fix error in this formula quite yet! For the purposes of teaching you features, let's pretend that we mistakenly decided E2's value was ok. Check the value off, but use the "I'm not sure" checkmark this time. *(Wait for mouse clicks).* Notice that the spreadsheet testedness bar is now up to 33% tested! *(circle with mouse)*

Before we fix the error in cell E2’s formula, I want to make sure that you all understand how “if formulas” work. Open up the formula for cell E2, the pen quantity check. The formula says “if the sum of the values in cell B2 and C2 is greater than 68, then the answer is the phrase "not enough pens", else the answer is the phrase “pen quantity ok"”.

The spreadsheets that you will be given might have errors in some of their formulas. And, sure enough, here’s an error in this spreadsheet: if we have 68 boxes of pens *or less*, then we do *not* have enough pens, otherwise the pen quantity is ok. So, change the “greater than” sign to a “less then sign, followed by an equal sign”. *(wait – people might get stuck on how to type in “<=”)* The formula now reads “if the sum of B2 and C2 is smaller than or equal to 68, then print the answer is "not enough pens", else the answer is "pen quantity ok"”. So far, our testedness bar is still at 33% and cell E2 has a purple border. Now, hit the “Apply” button.

What just happened?!? The testedness bar went down to 22% tested. The border color also changed – it went back to being red! And a question mark also appeared in the decision box, which used to contain a checkmark. Here’s why: since we changed the formula, the system had to discard some of our previous testing. After all, those tests were for the old formula. We have a new formula in this cell, so those tests are no longer valid. We now have to try some values on it to make sure that this new formula doesn’t contain errors.

Suppose we think the value in E2 now looks right. Check it off. *(Wait for mouse clicks.)* Notice that the percent testedness bar has gone back up to 33%.

The border of cell E2 turned purple. Hover over it. The tooltip says that the cell is 50% tested, and that it needs more testing. Remember what “testing” means? *(Pause)* We’re trying to get values that let us try different situations that we haven’t already tried.

Can you think of another situation that we haven’t tried for cell E2? *(Long Pause.)* How about looking at whether the value in cell E2 is correct when the pen quantity *is ok*? Change the value of C2 to 50. Don’t forget to hit “Apply”. A question mark appeared in the decision box of E2. Since we’re glad the answer is now “pen quantity ok”, check the value off for this situation. The cell border color is now blue, which means it’s 100% tested.

You were given two handouts for this task: a description of what the different cells should do and a set of example correct values. You will have both of these handouts for every task you get. Let’s pretend that, using those handouts, you decide that the value of D…5 is wrong, for the purposes of showing you more features. Cell D5 gives the discounted cost. Place an X-mark in D5’s decision box. To do this, first click the question mark, then pick one of the two x-marks. If you’re curious about any of the changes, don’t forget the tooltips are there to help you.

Look at the last sentence at the top of one of the handouts for PurchaseBudget; this is your task. It says, “Test the spreadsheet to see if it works correctly, and if you find any errors, fix them.”

Remember, if you are curious about any aspect of the system, you can hover your mouse over the item and read the tooltip. Also, you might find those checkmarks and X-marks to be useful. Since we have covered a lot of features in this tutorial, we have created a reference sheet to remind you of what we have covered. *(Hand out Reference Sheet)*

You may use it both as you explore your current spreadsheet, as well as for the actual tasks that you will perform after this tutorial. Remember that you also have handouts that tell you how the spreadsheet is supposed to behave and that give you example correct values. Starting now, you’ll have 3 minutes to test and explore the rest of this spreadsheet. While you’re exploring, look for errors in the spreadsheet and fix them. Once this time’s up, I’m going to show you one last feature.
(Wait 3 minutes – everyone walks around as assistants.)

Ok, the 3 minutes are up. Another feature that you might find useful to test and find errors is Help Me Test. Help Me Test comes up with suitable test values so you can make even more testing progress. Move your mouse cursor over the button that says “HELP ME TEST”, at the top of the spreadsheet. The tooltip reads “Help with testing this spreadsheet.” Click on it. You now get a couple more minutes to do any more exploring and error-fixing you want.

The tutorial is now over. Please minimize your spreadsheet.

(Hand out Gradebook description and sample values.)
Let's read the paragraph at the top of either one of your handouts:
"A teacher has updated a spreadsheet program that computes the course grade for his students. So far, he has only entered formulas for Sally. Once he is sure that those formulas are correct, he will also complete the rows for his other students. Your task is to help him by testing the updated spreadsheet and if you find any errors, fix them."

Now open the Gradebook spreadsheet by clicking on the bar labeled Gradebook at the bottom of the screen.

One of the handouts *(wave it around)* gives a description of how the spreadsheet should work. The other *(wave it around)* provides you with two correct sample report cards.

Remember, your task is to test the spreadsheet and, if you find any errors, fix them. To help you do this, use the checkmarks and x-marks by clicking cell decision boxes.

Start your task now, and I'll tell you when time is up.

(*Task is 22 minutes*)

Your time's up for this first task. Minimize your Gradebook spreadsheet.

(Hand out short Gradebook questionnaire.)

(Hand out Payroll description and example.)
Here is a payroll spreadsheet problem. Let's read the paragraph at the top of either one of your handouts:

"A spreadsheet that computes the net pay of three employees has been updated by one of your co-workers. So far, they have only entered the formulas for Bob. Once they are sure that those formulas are correct, they will go on to also modify the formulas for other employees. Your task is to test the updated spreadsheet and if you find any errors, fix them."

Now open the Payroll spreadsheet by clicking on the bar labeled Payroll at the bottom of the screen.

One of the handouts *(wave it around)* gives a description of how the spreadsheet should work. The other *(wave it around)* provides you with two correct sample payroll stubs.

Remember, your task is to test the spreadsheet, and if you find any errors, fix them. To help you do this, use the checkmarks and x-marks by clicking cell decision boxes.

Start your task now, and I'll tell you when time is up.

(*Task is 35 minutes*)

Your time's up for this final task. Minimize your Payroll spreadsheet.

(Hand out long Payroll questionnaire.) Please take your time to fill out this last questionnaire.

Thank you for participating in our study. Now if you hand us back your signed receipt you will receive your $20.

Appendix B: Questionnaires

Background Questionnaire

1. Gender (circle your selection): Male / Female
2. Age < 20 20 – 29 30 – 39 40 – 49 50 – 59 60+
3. Major or Educational Background: ______________________
4. School that you are attending:

   ____
5. Year or Degree Completed: Fresh. Soph. Jun. Sen. Post Bac. Grad.
6. Cumulative GPA: ______________________
7. Do you have previous programming experience?
   a. High school:
      - How many courses? _____
      - What programming languages? ________________________
   b. College:
      - How many courses? _____
      - What programming languages? ________________________
   c. Professional and/or recreational
      - How many years? _____
      - What programming languages? ________________________
8. Have you ever worked with formulas in spreadsheets for (please check all that apply):
   - ❑ A high school course How many spreadsheets? ____________
   - ❑ A college course How many spreadsheets? ____________

- ❑ Professional use How many years? _____________
- ❑ Personal use How many years? _____________

9. Have you participated in any previous Forms/3 experiments? Yes / No
10. Is English your primary language? Yes / No

If not, how long have you been speaking English? _______ years.

Pre-session Questionnaire

The following questions ask you to indicate whether you could use a new spreadsheet system under a variety of conditions. For each of the conditions please indicate whether you think you would be able to complete the job using the system.

Given a spreadsheet which performs common tasks (such as calculating course grades or payroll) I could find and fix errors:

... if there was no one around to tell me what to do as I go.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had never used a spreadsheet like it before.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had only the software manuals for references.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

... if I had seen someone else using it before trying it myself.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I could call someone for help if I got stuck.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if someone else had helped me get started.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had a lot of time to complete the task.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had just the built-in help facility for assistance.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if someone showed me how to do it first.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had used similar spreadsheets before this one to do this same task.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

Post-session Questionnaire (Payroll)

The following questions ask you to indicate whether you could use a new spreadsheet system under a variety of conditions. For each of the conditions please indicate whether you think you would be able to complete the job using the system.

Given a spreadsheet which performs common tasks (such as calculating course grades or payroll) I could find and fix errors:

... if there was no one around to tell me what to do as I go.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had never used a spreadsheet like it before.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had only the software manuals for references.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had seen someone else using it before trying it myself.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I could call someone for help if I got stuck.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if someone else had helped me get started.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had a lot of time to complete the task.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had just the built-in help facility for assistance.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if someone showed me how to do it first.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had used similar spreadsheets before this one to do this same task.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

Circle the answer corresponding to how much you agree or disagree with the following statements.

1. I am confident that I <u>found</u> all of the errors in the Payroll spreadsheet. (circle one)

Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

2. I am confident that I <u>fixed</u> all of the errors in the Payroll spreadsheet. (circle one)

Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

How much additional time would you need to complete this task?

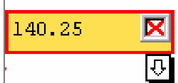
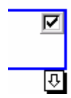
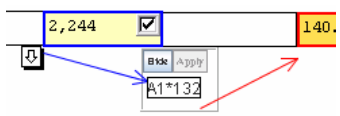
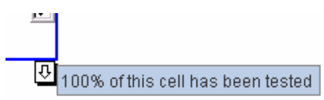
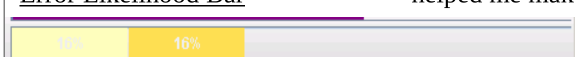
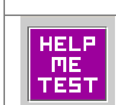
_____ None. It only took me _____ minutes.

_____ None. I took about the entire time.

_____ I would need about _____ more minutes.

_____ I am not sure.

3. Mark how you found the following features that were available to you for **finding and fixing errors**:

Cell Border Colors helped me make progress 	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
 Interior Cell Coloring (yellow and orange) helped me make progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
 X-Marks helped me make progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Checkmarks (√) helped me make progress 	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Arrows helped me make progress 	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Tooltips helped me make progress 	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Percent Testedness Bar helped me  make progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Error Likelihood Bar helped me make  progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
 "Help Me Test" helped me make progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Spreadsheet Description (on the handout) helped me make progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

Sample Values (on the handout) helped me make progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

4. From the following list of features, pick the two you preferred the most and describe how you used them to find and/or fix errors:
Cell Border Colors, Interior Cell Colorings, X-Marks, Checkmarks, Arrows, Tooltips, Percent TestednessBar, Error Likelihood Bar, Help Me Test

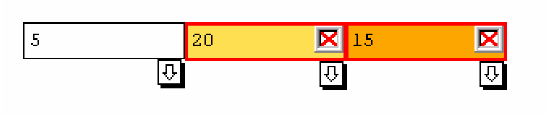
A) Most Preferred Feature: ____
How you used it to find and/or fix errors:

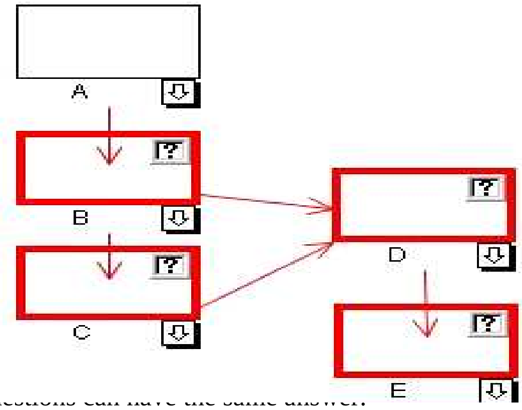
B) Second Most Preferred Feature: ____
How you used it to find and/or fix errors:

5. In the figure below, what does the X-mark in the decision box mean?

6. In the figure below, what does the orange color in the interior of the cell mean?

7. In the figure below, what does it mean when the color in the interior of one cell is a darker orange than others?





answers from the choices below. One or more questions can have the same answer.

8. If we place an X-mark in cell D, the color of the interior of cell D:
   a. Remains the same
   b. Gets darker
   c. Gets lighter
   d. Don't know

9. If we place an X-mark in cell D, the color of the interior of cell C:
   a. Remains the same
   b. Gets darker
   c. Gets lighter
   d. Don't know

10. If we place an X-mark in cell D, the color of the interior of cell E:
   a. Remains the same
   b. Gets darker
   c. Gets lighter
   d. Don't know

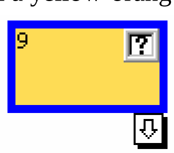
Assume for the next three Questions (13-15) that an X-mark has been placed on the cell D.

11. If we place an X-mark in cell C, the color of the interior of cell C:
   a. Remains the same
   b. Gets darker
   c. Gets lighter
   d. Don't know

12. If we place an X-mark in cell C, the color of the interior of cell B:
   a. Remains the same
   b. Gets darker
   c. Gets lighter
   d. Don't know

13. If we place a Checkmark in cell C, the color of the interior of cell D:
   a. Remains the same
   b. Gets darker
   c. Gets lighter
   d. Don't know

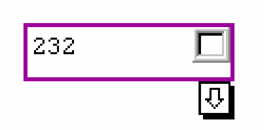
14. What does a blue border of a cell with a yellow-orange interior mean (refer to figure below)?



(Circle 1 option for each part)

a) The value is: (circle 1)	CORRECT	WRONG	COULD BE EITHER
b) The cell is: (circle 1)	TESTED	UNTESTED	COULD BE EITHER
c) The cell has: (circle 1)	ERROR LIKELIHOOD	NO ERROR LIKELIHOOD	COULD BE EITHER
d) My answers to a, b, and c are just guesses.	YES, JUST GUESSES	NO, NOT GUESSES	SOME YES, SOME NO
e) The combination of blue border and yellow-orange interior colors on this cell: (circle 1)	MAKES SENSE	MAKES NO SENSE	NOT SURE

15. There is a cell with a purple border and a **blank** in its decision box.



A) If you place a checkmark in that decision box, does the border color change?

______ yes
______ no
______ I'm not sure

B) What is the border color after you've placed the checkmark in the decision box?
______ Red
______ Same purple
______"Bluer" purple
______ Blue
______ Depends on the formula, but definitely more red
______ Depends on the formula, but definitely more blue
______ I’m not sure

C) If you place a checkmark in the decision box, the form's Percent Tested Bar, shown below, will:

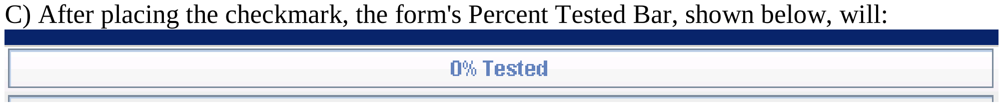
______Increase
______Stay the same
______Decrease
______Not enough information to tell
______I'm not sure

16. There is a cell with a purple border and a **question mark** in its decision box.

A) If you place a checkmark in that decision box, does the border color change?
______ Yes
______ No
______ I'm not sure

B) What is the border color after you have placed the checkmark in the decision box?
______Red
______Same purple
______"Bluer" purple
______ Blue
______ Depends on the formula, but definitely more red
______ Depends on the formula, but definitely more blue
______I'm not sure

C) After placing the checkmark, the form's Percent Tested Bar, shown below, will:



______Increase
______Stay the same
______Decrease
______Not enough information to tell
______I'm not sure

17. In a few sentences, describe your general strategy for finding and fixing errors in the spreadsheets.

18. Did you place X-marks? If yes answer Question 18A, otherwise answer Question 18B.
A) When I placed an X-mark…

… I was afraid that I would not use them properly.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
… I was afraid I would take too long to learn them.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

B) I did not place X-marks because…

… I was afraid that I would not use them properly.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
… I was afraid I would take too long to learn them.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

19. Did you use Help Me Test? If yes answer Question 19A, otherwise answer Question 19B.
A) When I used Help Me Test…

… I was afraid that I would not use it properly.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
… I was afraid I would take too long to learn it.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

B) I did not use Help Me Test because…

… I was afraid that I would not use it properly.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
… I was afraid I would take too long to learn it.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

Thank you for participating in our study!

Post-session Questionnaire (Gradebook)

The following questions ask you to indicate whether you could use a new spreadsheet system under a variety of conditions. For each of the conditions please indicate whether you think you would be able to complete the job using the system.

Given a spreadsheet which performs common tasks (such as calculating course grades or payroll) I could find and fix errors:

... if there was no one around to tell me what to do as I go.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had never used a spreadsheet like it before.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had only the software manuals for references.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had seen someone else using it before trying it myself.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I could call someone for help if I got stuck.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if someone else had helped me get started.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had a lot of time to complete the task.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had just the built-in help facility for assistance.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if someone showed me how to do it first.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
... if I had used similar spreadsheets before this one to do this same task.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

Circle the answer corresponding to how much you agree or disagree with the following statements.

3. I am confident that I found all of the errors in the Gradebook spreadsheet. (circle one)

Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

4. I am confident that I fixed all of the errors in the Gradebook spreadsheet. (circle one)

Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

How much additional time would you need to complete this task?

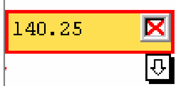
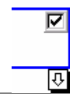
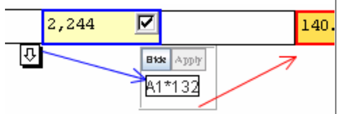
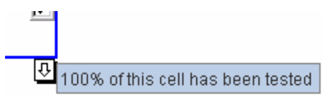
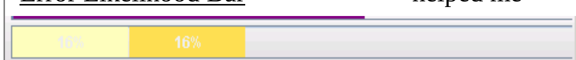
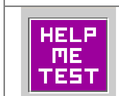
_____ None. It only took me _____ minutes.

_____ None. I took about the entire time.

_____ I would need about _____ more minutes.

_____ I am not sure.

3. Mark how you found the following features that were available to you for **finding and fixing errors**:

Cell Border Colors helped me make progress 	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
 Interior Cell Coloring (yellow and orange) helped me make progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
 X-Marks helped me make progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Checkmarks (√) helped me make progress 	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Arrows helped me make progress 	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Tooltips helped me make progress 	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Percent Testedness Bar helped  me make progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Error Likelihood Bar helped me  make progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
 "Help Me Test" helped me make progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Spreadsheet Description (on the handout) helped me make progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
Sample Values (on the handout) helped me make progress	Never Used this feature	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

4. From the following list of features, pick the two you preferred the most and describe how you used them to find and/or fix errors:
Cell Border Colors, Interior Cell Colorings, X-Marks, Checkmarks, Arrows, Tooltips, Percent TestednessBar, Error Likelihood Bar, Help Me Test

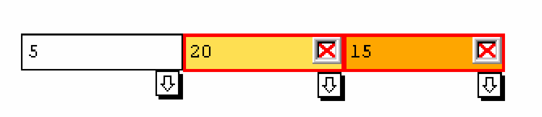
A) Most Preferred Feature: ____
How you used it to find and/or fix errors:

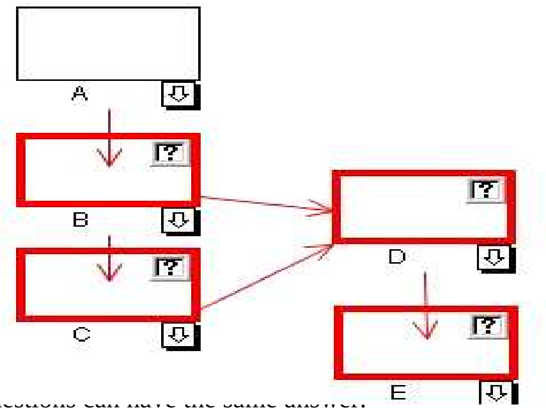
B) Second Most Preferred Feature: ____
How you used it to find and/or fix errors:

5. In the figure below, what does the X-mark in the decision box mean?

6. In the figure below, what does the orange color in the interior of the cell mean?

7. In the figure below, what does it mean when the color in the interior of one cell is a darker orange than others?





answers from the choices below. One or more questions can have the same answer.

8. If we place an X-mark in cell D, the color of the interior of cell D:
   a. Remains the same
   b. Gets darker
   c. Gets lighter
   d. Don't know

9. If we place an X-mark in cell D, the color of the interior of cell C:
   a. Remains the same
   b. Gets darker
   c. Gets lighter
   d. Don't know

10. If we place an X-mark in cell D, the color of the interior of cell E:
   a. Remains the same
   b. Gets darker
   c. Gets lighter
   d. Don't know

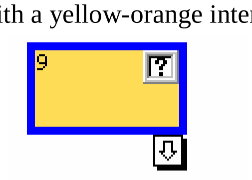
Assume for the next three Questions (13-15) that an X-mark has been placed on the cell D.

11. If we place an X-mark in cell C, the color of the interior of cell C:
   a. Remains the same
   b. Gets darker
   c. Gets lighter
   d. Don't know

12. If we place an X-mark in cell C, the color of the interior of cell B:
   a. Remains the same
   b. Gets darker
   c. Gets lighter
   d. Don't know

13. If we place a Checkmark in cell C, the color of the interior of cell D:
   a. Remains the same
   b. Gets darker
   c. Gets lighter
   d. Don't know

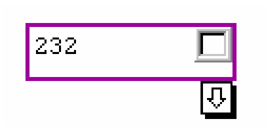
14. What does a blue border of a cell with a yellow-orange interior mean (refer to figure below)?



(Circle 1 option for each part)

a) The value is: (circle 1)	CORRECT	WRONG	COULD BE EITHER
b) The cell is: (circle 1)	TESTED	UNTESTED	COULD BE EITHER
c) The cell has: (circle 1)	ERROR LIKELIHOOD	NO ERROR LIKELIHOOD	COULD BE EITHER
d) My answers to a, b, and c are just guesses.	YES, JUST GUESSES	NO, NOT GUESSES	SOME YES, SOME NO
e) The combination of blue border and yellow-orange interior colors on this cell: (circle 1)	MAKES SENSE	MAKES NO SENSE	NOT SURE

15. There is a cell with a purple border and a **blank** in its decision box.



A) If you place a checkmark in that decision box, does the border color change?

______ yes
______ no
______ I'm not sure

B) What is the border color after you've placed the checkmark in the decision box?

______ Red
______ Same purple
______"Bluer" purple
______ Blue
______ Depends on the formula, but definitely more red
______ Depends on the formula, but definitely more blue
______ I’m not sure

C) If you place a checkmark in the decision box, the form's Percent Tested Bar, shown below, will:

______Increase
______Stay the same
______Decrease
______Not enough information to tell
______I'm not sure

16. There is a cell with a purple border and a **question mark** in its decision box.

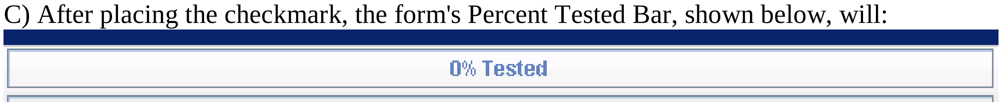
A) If you place a checkmark in that decision box, does the border color change?

______ Yes
______ No
______ I'm not sure

B) What is the border color after you have placed the checkmark in the decision box?

______Red
______Same purple
______"Bluer" purple
______ Blue
______ Depends on the formula, but definitely more red
______ Depends on the formula, but definitely more blue
______I'm not sure

C) After placing the checkmark, the form's Percent Tested Bar, shown below, will:



______Increase
______Stay the same
______Decrease
______Not enough information to tell
______I'm not sure

17. In a few sentences, describe your general strategy for finding and fixing errors in the spreadsheets.

18. Did you place X-marks? If yes answer Question 18A, otherwise answer Question 18B.
A) When I placed an X-mark…

… I was afraid that I would not use them properly.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
… I was afraid I would take too long to learn them.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

B) I did not place X-marks because…

… I was afraid that I would not use them properly.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
… I was afraid I would take too long to learn them.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

19. Did you use Help Me Test? If yes answer Question 19A, otherwise answer Question 19B.
A) When I used Help Me Test…

… I was afraid that I would not use it properly.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
… I was afraid I would take too long to learn it.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

B) I did not use Help Me Test because…

… I was afraid that I would not use it properly.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
… I was afraid I would take too long to learn it.	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree

Thank you for participating in our study!

Appendix C: Spreadsheets and Spreadsheet Descriptions

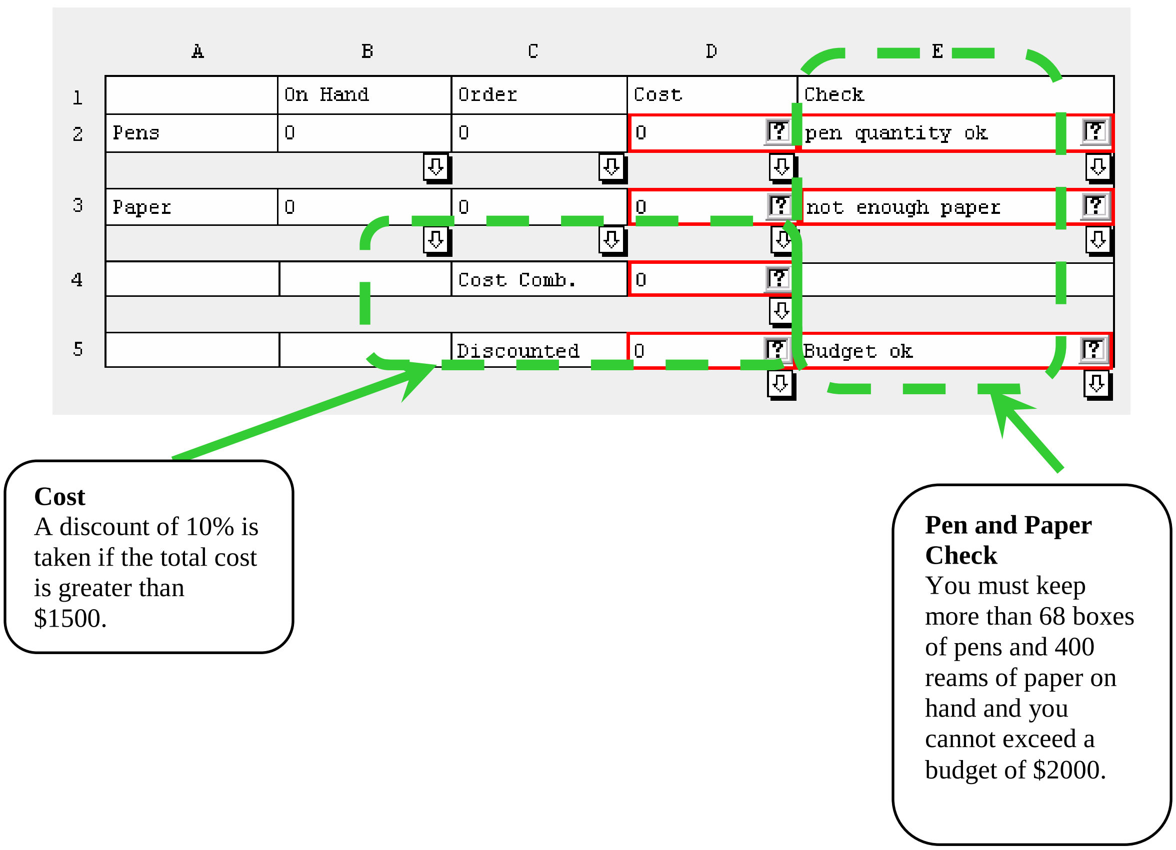


Figure 9. Description of Purchase Budget Spreadsheet

EXAMPLE CORRECT VALUES FOR PURCHASE BUDGET SPREADSHEET PROBLEM
You are in charge of ordering office supplies for the office you work at. You must order enough pens and paper to have on hand, but you cannot spend more than your allotted budget for office supplies. Test the spreadsheet and if you find any errors, fix them.

	A	B	C	D	E
1		On Hand	Order	Cost	Check
2	Pens	75	0	0	pen quantity ok
	Paper	21	400	1,600	paper quantity ok
4			Cost Comb.	1,600	
5			Discounted	1,440	Budget ok

Figure 10. Example of Purchase Budget Spreadsheet

Description of Payroll Task

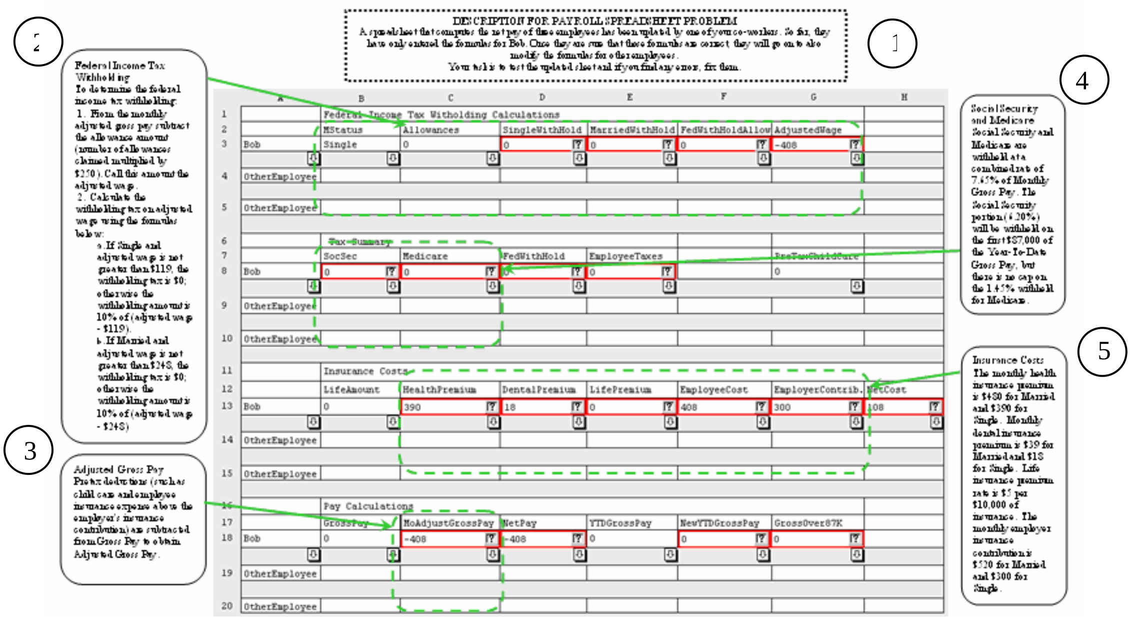


Figure 11. Miniaturized description of Payroll spreadsheet

The description sheet has been miniaturized because of space constraints. The texts in the boxes have been expanded below:

1

DESCRIPTION FOR PAYROLL SPREADSHEET PROBLEM

A spreadsheet that computes the net pay of three employees has been updated by one of your co-workers. So far, they have only entered the formulas for Bob. Once they are sure that those formulas are correct, they will go on to also modify the formulas for other employees.

Your task is to test the updated sheet and if you find any errors, fix them.

2

Federal Income Tax Withholding

To determine the federal income tax withholding:

1. From the monthly adjusted gross pay subtract the allowance amount (number of allowances claimed multiplied by $250). Call this amount the adjusted wage.

2. Calculate the withholding tax on adjusted wage using the formulas below:

- **a.** If Single and adjusted wage is not greater than $119, the withholding tax is $0; otherwise the withholding amount is 10% of (adjusted wage - $119).
- **b.** If Married and adjusted wage is not greater than $248, the withholding tax is $0; otherwise the withholding amount is 10% of (adjusted wage - $248)

3

Adjusted Gross Pay

Pretax deductions (such as child care and employee insurance expense above the employer's insurance contribution) are subtracted from Gross Pay to obtain Adjusted Gross Pay.

4

Social Security and Medicare

Social Security and Medicare are withheld at a combined rate of 7.65% of Monthly Gross Pay. The Social Security portion (6.20%) will be withheld on the first $87,000 of the Year-To-Date Gross Pay, but there is no cap on the 1.45% withheld for Medicare.

5

Insurance Costs

The monthly health insurance premium is $480 for Married and $390 for Single. Monthly dental insurance premium is $39 for Married and $18 for Single. Life insurance premium rate is $5 per $10,000 of insurance. The monthly employer insurance contribution is $520 for Married and $300 for Single.

Payroll Example Sheet

EXAMPLE CORRECT VALUES FOR PAYROLL SPREADSHEET PROBLEM

A spreadsheet that computes the net pay of three employees has been updated by one of your co-workers. So far, they have only entered the formulas for Bob. Once they are sure that those formulas are correct, they will go on to also modify the formulas for other employees. Your task is to test the updated spreadsheet and if you find any errors, fix them.

	A	B	C	D	E	F	G	H
1		Federal Income Tax Witholding Calculations						
2		MStatus	Allowances	SingleWithHold	MarriedWithHold	FedWithHoldAllow	AdjustedWage	
3	Bob	Single	1					
4	OtherEmployee							
5	OtherEmployee							
6		Tax Summary						
7		SocSec	Medicare	FedWithHold	EmployeeTaxes		PreTaxChildCare	
8	Bob	372	87	551.8	1,010.8		0	
9	OtherEmployee							
10	OtherEmployee							
11		Insurance Costs						
12		LifeAmount	HealthPremium	DentalPremium	LifePremium	EmployeeCost	EmployerContrib.	NetCost
13	Bob	10,000	390	18	5	413	300	113
14	OtherEmployee							
15	OtherEmployee							
16		Pay Calculations						
17		GrossPay	MoAdjustGrossPay	NetPay	YTDGrossPay	NewYTDGrossPay	GrossOver87K	
18	Bob	6,000	5,887	4,876.2	54,000			
19	OtherEmployee							
20	OtherEmployee							

Figure 12. Example sheet of Payroll spreadsheet

Description of Gradebook Task

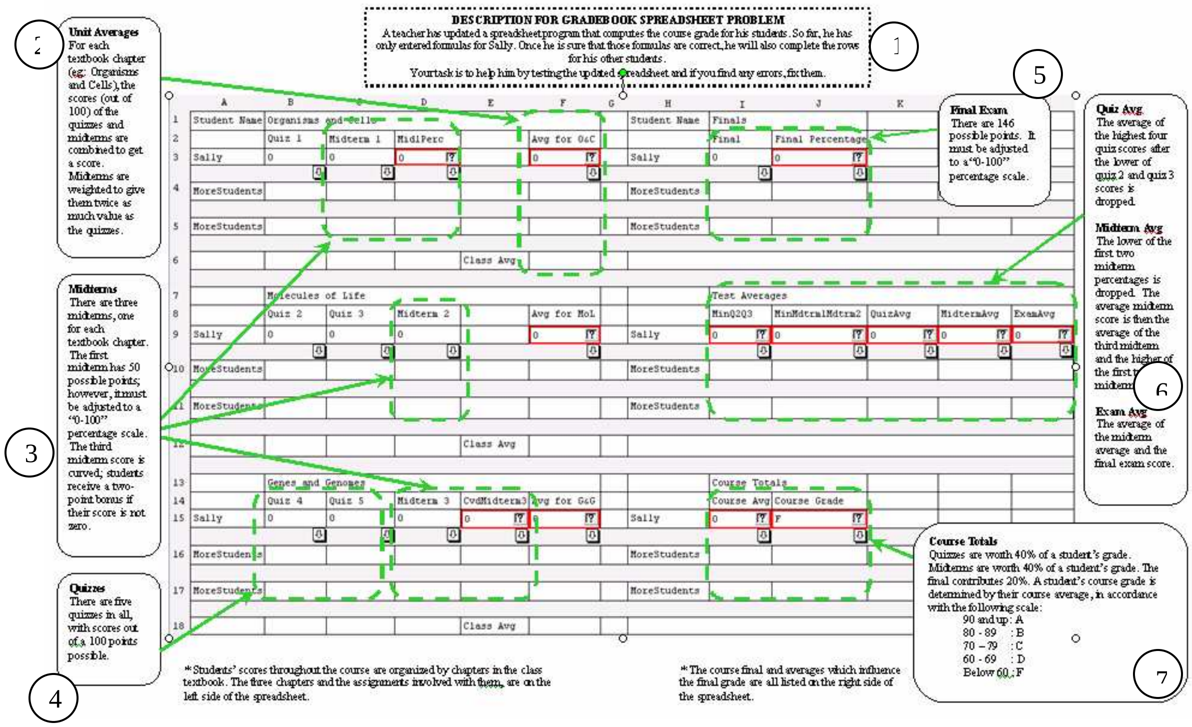


Figure 13. Miniaturized description of Payroll spreadsheet

The description sheet has been miniaturized because of space constraints. The texts in the boxes have been expanded below:

1

DESCRIPTION FOR GRADEBOOK SPREADSHEET PROBLEM
A teacher has updated a spreadsheet program that computes the course grade for his students. So far, he has only entered formulas for Sally. Once he is sure that those formulas are correct, he will also complete the rows for his other students.
Your task is to help him by testing the updated spreadsheet and if you find any errors, fix them.

2 →

Unit Averages
For each textbook chapter (eg: Organisms and Cells), the scores (out of 100) of the quizzes and midterms are combined to get a score. Midterms are weighted to give them twice as much value as the quizzes

3 →

Midterms
There are three midterms, one for each textbook chapter.The first midterm has 50 possible points; however, it must be adjusted to a “0-100” percentage scale. The third midterm score is curved; students receive a two-point bonus if their score is not zero.

4 →

Quizzes
There are five quizzes in all, with scores out of a 100 points possible.

5 →

Final Exam
There are 146 possible points. It must be adjusted to a "0-100" percentage scale.

Gradebook Example Sheet

EXAMPLE CORRECT VALUES FOR GRADEBOOK SPREADSHEET PROBLEM
A teacher has updated a spreadsheet program that computes the course grade for his students. So far, he has only entered formulas for Sally. Once he is sure that those formulas are correct, he will also complete the rows for his other students.
Your task is to help him by testing the updated spreadsheet and if you find any errors, fix them

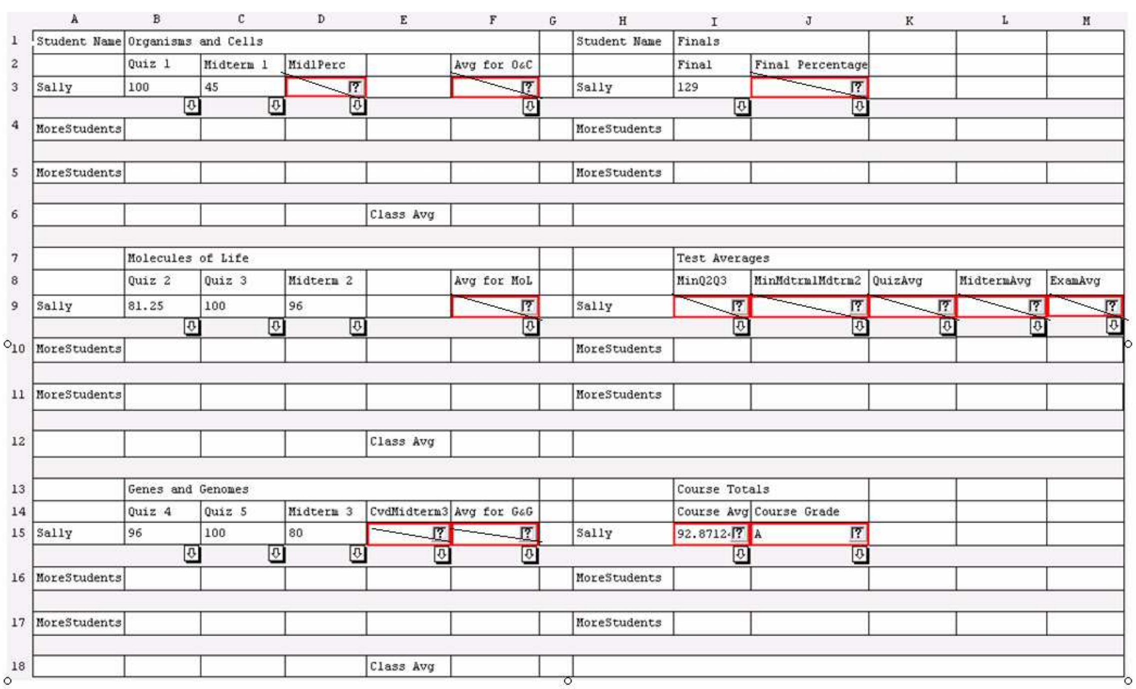


Figure 14. Example sheet of Gradebook spreadsheet