

«ОЦЕНКА РАВНЫХ» КАК МЕТОД ЭКСПЕРТНОЙ ОЦЕНКИ: ОПЫТ ПРИМЕНЕНИЯ В ОБЛАСТИ COMPUTER SCIENCE

М.А. Плаксин

*Национальный исследовательский университет Высшая школа экономики
(Пермский филиал)*

Россия, 614070, г. Пермь, ул. Студенческая, 38

E-mail: mapl@list.ru

Аннотация: «Оценка равных» – один из методов экспертной оценки, в котором в качестве экспертов выступают лица «равные по рангу» оцениваемому лицу. В докладе рассматривается суть метода и его применение для оценки качества компьютерных программ и совершенствования образовательного процесса в вузе. Описываются экспериментальные исследования, проведенные в разное время на механико-математическом факультете Пермского государственного НИУ и факультете бизнес-информатики Пермского филиала НИУ Высшая школа экономики.

Введение

Экспертная оценка является одним из основных методов обоснования решений во многих плохо формализованных областях. Применение этого метода связано с рядом сложностей. Одна из них заключается в том, что мнение эксперта всегда в той или иной степени субъективно. (Там, где можно применить формальные методы, отсекающие субъективизм, экспертная оценка просто не нужна.) Для уменьшения субъективности можно привлечь к экспертизе не одного человека, а сразу нескольких (организовать консилиум). Но для этого придется

- 1) найти достаточное количество людей, хорошо разбирающихся в анализируемой проблеме;
- 2) организовать их совместную деятельность;
- 3) оплатить эту деятельность.

В качестве одного из наиболее известных вариантов такой совместной экспертизы можно назвать метод Дельфи [1]. Применение метода Дельфи хорошо демонстрирует сложности такой процедуры: высокую трудоемкость, большие затраты времени, высокую стоимость, сложности с формированием команды экспертов.

В данной статье описывается опыт применения другого метода совместной экспертной оценки – метода «оценки равных» («peer assessment»). Этот метод

предназначен для оценки квалификации отдельных специалистов. При этом в роли «оценщиков» выступают коллеги оцениваемого сотрудника «равные ему по рангу». Такой подход к формированию группы экспертов сразу снимает проблемы, связанные с поиском высокоуровневых специалистов и (соответственно) высокой оплатой их труда. Но при этом возникают вопросы, связанные с качеством результатов экспертизы. Принято считать, что квалификация оценивающего обязательно должна быть выше квалификации оцениваемого, поскольку иначе оценивающий не сможет достоверно определить, где именно оцениваемый допустил ошибки, в чем они заключались, насколько они серьезны. Опыт применения «оценки равных» показывает, что группа «равных по рангу» при соблюдении ряда условий способна выработать достаточно точную оценку деятельности своего коллеги.

Противоположностью оценки «равными по рангу» является «оценка начальником». Заметим, что по сравнению с «оценкой начальником» «оценка равных» имеет следующее психологическое обоснование. Для любого подчиненного нормально стремиться произвести на начальника хорошее впечатление, показать себя с лучшей стороны, иногда даже преувеличить собственные достоинства. «пустить пыль в глаза». Поэтому начальник

зачастую видит своих подчиненных не такими, каковы они на самом деле, а такими, которыми они стараются ему понравиться. В таких условиях окажется, что оценка, даваемая начальником, основывается на неверных исходных данных. Подчиненный имеет возможность «пустить начальнику пыль в глаза», поскольку начальник видит его относительно редко. И именно в эти редкие минуты надо постараться и произвести нужное впечатление. В общении с коллегами, равными по рангу, такое невозможно. С ними человек общается постоянно.

Интерес автора доклада к теме «оценки равных» был вызван публикацией в [3] сообщения об экспериментах, проведенных Беном Шейдерманом в 1977-1978 гг. в ряде программистских фирм США. В 1994 г. эти эксперименты были по возможности точно воспроизведены на механико-математическом факультете Пермского госуниверситета [2]. Результаты оказались настолько интересны, что процедура в течение ряда лет (до изменения в программе курса) проводилась в рамках учебного процесса студентов-программистов. В 2014-15 уч. г. после многолетнего перерыва процедура «оценки равных» была проведена на факультете бизнес-информатики Пермского филиала НИУ Высшая школа экономики. Естественно, в процесс оценки были внесены корректировки, отражающие изменения в условиях труда программистов и в условиях обучения студентов, происшедшие за это время. Был коренным образом переработан список оценочных вопросов. Работа с бумажными документами была заменена работой с softcopy, очная процедура оценки по принципу «всем в одно время в одном месте» – работой через Интернет в любое удобное время в пределах заданного срока.

Перенос «оценки равных» из промышленных фирм в учебное заведение придал процедуре новую направленность. Теперь она приобрела не только (да и не столько) оценочный характер, но и обучающий.

Данный доклад обобщает материалы, полученные в исследованиях 90-х гг. и в последнем эксперименте 2014-15 уч.г.

1. Описание процедуры «оценки равных» для оценки качества компьютерных программ

В исследованиях Шнейдермана и выстроенных на их базе процедурах в Пермском университете и Пермском филиале ВШЭ процесс оценки «равными по рангу» выглядит следующим образом. Формируется группа программистов, имеющих примерно одинаковую подготовку и опыт работы. Каждый из них представляет для оценки одну свою программу. Из программы удаляются любые указания на авторство. После чего каждый рецензент получает свой экземпляр программы (когда-то – в виде распечатки, сейчас – по e-mail.) Оценка осуществляется по специальному бланку, содержащему ряд вопросов по каждой программе в отдельности, сразу по всем программам и по процедуре рецензирования. Ответы на вопросы можно дополнять любыми комментариями. Рецензии сдаются администратору эксперимента, который готовит отчеты для каждого из участников. Отчеты включают оценки и комментарии, данные рецензентами (без указания имени рецензента), некоторую обобщающую информацию по всей группе, а также ряд вычисляемых параметров, предназначенных для оценки достоверности мнения экспертов. Отчет готовится индивидуально для каждого участника процедуры. (в 70-е и 90-е гг. XX в. на бумаге, сейчас – в файле).

В идеале вся оценка проводится строго анонимно. Рецензенты не знают, чьи программы они оценивают, рецензируемый не знает, кто оценивал его программы. Все программы, отсылаемые на рецензирование, известны только по номерам. Все оценки в отчетах – анонимны. В отчете, предназначенном конкретному программисту, указывается, какая из оцениваемых программ является его программой и какие из приведенных в отчете оценок проставлены им.

2. Детальное обсуждение процедуры рецензирования

Уточним некоторые детали вышеописанной схемы.

«Равный ранг» ранг рецензентов достигался тем, что в каждой процедуре

принимали участие студенты одного курса и одной специальности («прикладная математика» в ПГУ и «программная инженерия» в ВШЭ). Это дает дополнительную возможность сравнить оценки студентов-младшекурсников, студентов-старшекурсников и профессиональных программистов, с которыми экспериментировал Шнейдерман.

Оптимальной по размеру представляется группа в пять человек. С такими группами экспериментировал Шнейдерман. В наших исследованиях размер группы варьировался от трех до шести человек. Опыт показывает, что при меньших группах через чур возрастает роль случайных отклонений в оценках, при больших – увеличивается нагрузка на рецензента и, соответственно, уменьшается внимание, которое он может уделить каждой программе. Кроме того, в случае, когда размер группы равен пяти, удается достаточно легко получить ответы на некоторые вопросы, не задавая этих вопросов «в лоб». В больших группах сделать это много сложнее.

В «бумажные» времена играл роль вопрос о качестве печати. Сегодня он превратился в вопрос о допустимых форматах файлов. Поскольку речь идет о текстах программ, можно обойтись текстовыми файлами. Но можно использовать и DOC'и.

Размер программ в экспериментах Шнейдермана варьировался от 50 до 650 строк, в наших – от 200 до 1000 строк. Размер будет зависеть от используемого языка и системы программирования. Поскольку в НИУ ВШЭ обучение ведется на C#, этот язык и использовался в исследовании. Для C# оптимальным представляется размер в 4-6 страниц.

В «бумажные» времена приходилось регулировать время рецензирования. В экспериментах Шнейдермана на оценку одной программы давалось 35-45 минут. Был запрещен переход к следующей программе до истечения этого времени, а также возврат к уже оцененным программам. В ПГУ все программы выдавались на руки сразу, порядок работы не регламентировался, но вся процедура строилась таким образом, чтобы уложиться в одну университетскую пару (два академических часа и перерыв между «ими»). При работе через Интернет никаких

ограничений на время оценки не накладывалось (кроме требования прислать заполненную анкету к указанному сроку).

Надо отметить, что время оценки существенно зависит от количества и сложности вопросов в анкете.

«Бумажные» эксперименты проводились в аудитории достаточно большой, чтобы участников можно было рассадить на некотором удалении друг от друга. Рецензентам настоятельно рекомендовалось работать самостоятельно и не обсуждать программы с коллегами и авторами. Сейчас эти рекомендации сохраняются, но проконтролировать их исполнение невозможно.

Для обеспечения анонимности принимается ряд мер. Состав «пятерок» – групп студентов, оценивающих программы друг друга – известен только администратору. Все программы обозначаются некими кодами. В заключительных отчетах код программы заменяется на номер (по существу, другой код). Имена рецензентов не указываются вообще. Комментарии копируются в файл отчета без указания автора. Отчеты готовятся персонально для каждого участника, для каждого указывается номер его программы и оценки, проставленные именно им. Несмотря на все старания, анонимность удавалось гарантировать только по отношению к рецензентам, но не по отношению к авторам программ. Однокурсники, как правило, достаточно хорошо знают работы друг друга. Поэтому принимались специальные меры для того, чтобы у рецензентов не возникало желания зависеть или занизить оценку. В частности, при оценивании преподавателем работы студента в данном процессе во внимание принимались только три вещи: сам факт участия, исполнение регламента (соблюдение предписанных требований) и подробность комментариев, оставленных рецензентом. Оценки, полученные рецензируемой программой, в расчет не брались.

Отчеты рекомендуется готовить как можно скорее.

Анкета, по которой проводится рецензирование, состоит из четырех частей.

Первая часть бланка – это группа субъективных вопросов о наличии или отсутствии у программы того или иного свойства. В «бумажном» варианте их было

всего 13. В «файловом» варианте их количество было увеличено до 43. Оценка дается отдельно для каждой программы по семибальной шкале (1 – «нет», 7 – «да»). Список вопросов может изменяться.

Вторая часть вопросника – ранжирование всех отрецензированных программ от наилучшей до наихудшей. задание отранжировать программы дается не «в лоб», а косвенно. Рецензенту предлагается указать сначала самую хорошую программу, потом – самую плохую и, наконец, программу, которую он бы поставил на второе место. Именно здесь используется тот факт, что на каждого рецензента приходится четыре программы.

Третья часть анкеты – пять вопросов о процессе рецензирования равными по рангу, ответы на которые также следует давать по шкале от 1 до 7:

1. Узнали ли Вы в ходе рецензирования что-либо полезное о стиле программирования?

2. Изменили бы Вы стиль написания Ваших программ, если бы Вам сказали, что рецензирование равными по рангу будет проводиться каждые 6 месяцев?

3. Считаете ли Вы, что процесс рецензирования равными по рангу может оказаться эффективным средством улучшения программирования в Вашей организации?

4. Считаете ли Вы, что организаторы рецензирования сделали все возможное для сохранения анонимности?

5. Ваше мнение по поводу процедуры рецензирования равными по рангу и предложения по ее усовершенствованию?

Последняя четвертая часть анкеты – четыре вопроса, направленные на совершенствование набора критериев оценки программ:

1. Укажите, пожалуйста, два вопроса, ответ на которые вызвал наибольшие затруднения.

2. Если затруднения вызваны нечеткой или некорректной формулировкой этих вопросов, не могли бы Вы предложить для них свою формулировку, более четкую и корректную.

3. Укажите, пожалуйста, два вопроса, наименее полезные с Вашей точки зрения для оценивания качества программы.

4. Назовите, пожалуйста, темы или вопросы, важные по Вашему мнению, но недостаточно отраженные в заполненном Вами бланке:

Далее приведен фрагмент бланка анкеты.

РЕЦЕНЗИРОВАНИЕ РАВНЫМИ ПО РАНГУ

ИНСТРУКЦИЯ: Укажите Вашу фамилию или идентификационный номер в поле "Рецензирующий", затем укажите коды (номера) рецензируемых программ, после этого ответьте, пожалуйста, на вопросы, указывая оценку по шкале от 1 до 7: НЕТ: 1, – ДА: 7
Перед постановкой оценок следует ознакомиться с текстом оцениваемых программ.

Рецензирующий: _____

Замечание: Приведенные примеры написаны на языке Паскаль. При оценке Вам надо переложить их на язык программирования оцениваемой программы.

СУБЪЕКТИВНЫЕ ВОПРОСЫ	№ программы			
1. Программа оформлена в одном стиле. Если в каких-то правилах допускается вариативность, то всегда применяется какой-то один вариант.				
Переменные. Имена (идентификаторы)				
2. Любая переменная используется в одном единственном смысле.				
3. Смысл переменных, подпрограмм, типов и пр. отражен в их имени (идентификаторе).				

Оценка равных. 21.03.96. 2 курс, 1 гр., 2 звездочка.

Вопрос	Программа 2				Программа 5				Групповое среднее								
	Согл.	Среднее	Дисперс	Разброс	Согл.	Среднее	Дисперс	Разброс									
1. Хороши ли имена переменных?	6	6	7	6	4	6.25	0.1875	1	7	5	7	6	3	6.25	0.6875	2	5.7
2. Полезные и достаточные комментарии?	7	4	7	7	3	6.25	1.6875	3	2	1	1	2	4	1.5	0.25	1	4.5
3. Пропуски строк и пробелы?	4	5	7	4	3	5	1.5	3	4	5	7	4	3	5	1.5	3	4.85
4. Понятна ли логика нижнего уровня?	6	4	7	7	3	6	1.5	3	1	3	1	3	2	2	1	2	5
5. Схема проектирования верхнего уровня?	7	4	7	7	3	6.25	1.6875	3	7	4	7	5	2	5.75	1.6875	3	6.35
6. Хорош ли алгоритм?	6	5	7	3	2	5.25	2.1875	4	5	6	7	6	3	6	0.5	2	5.7
7. Легко ли понять программу в целом?	6	5	7	7	3	6.25	0.6875	2	6	2	1	5	2	3.5	4.25	5	5.35
8. Легко модифицировать?	4	3	7	6	2	5	2.5	4	3	3	1	4	3	2.75	1.1875	3	4.5
9. Не зависит от компилятора?	2	4	3	6	2	3.75	2.1875	4	1	3	1	6	2	2.75	4.1875	5	3.65
10. Машинно-независима?	6	3	1	6	2	4	4.5	5	3	3	7	5	2	4.5	2.75	4	5.2
11. Были бы Вы горды, написав эту пр-му?	7	6	7	7	4	6.75	0.1875	1	7	6	1	6	3	5	5.5	6	4.5
12. Разумно ли используются структуры данных?	7	5	7	7	3	6.5	0.75	2	7	7	7	7	4	7	0	0	6.65
13. Легко ли отлаживать эту программу?	5	4	6	4	3	4.75	0.6875	2	2	4	1	4	2	2.75	1.6875	3	4.8
						Согл.	Среднее	Дисперс	Разброс	Средняя дисперсия по звездочке							
Программа 2 получила оценки	2	4	1	1	3	2	1.5	3	1.9144								
Программа 5 получила оценки	1	2	4	3	2	2.5	1.25	2	Средний разброс:								
Программа 8 получила оценки	4	3	2	4	3	3.25	0.6875	2	3								
Программа 11 получила оценки	2	1	3	2	3	2	0.5	2	Средняя согласованность								
Программа 14 получила оценки	3	4	1	3	3	2.75	1.1875	3	3.3692								
Результаты заключительных оценок:						Согл.	Среднее	Дисперс	Разброс	Скорее да	Скорее нет						
1. Узнали ли полезное о стиле?	7	7	7	1	7	4	5.8	5.76	6	(>=5)	(=<3)						
2. Изменили бы стиль при регулярной оценке?	1	7	1	6	4	2	3.8	6.16	6	4	1						
3. Полезно для улучшения пр-ия в организации?	5	7	1	7	7	3	5.4	5.44	6	2	2						
4. Для сохранения анонимности сделано все?	5	4	3	6	5	3	4.6	1.04	3	3	1						

Оценка равных. 21.03.96. 2 курс, 1 гр., 2 звездочка.

Вопрос	Программа 8				Программа 11				Групповое среднее								
	Согл.	Среднее	Дисперс	Разброс	Согл.	Среднее	Дисперс	Разброс									
1. Хороши ли имена переменных?	5	7	1	6	2	4.75	5.1875	6	5	7	5	6	3	5.75	0.6875	2	5.7
2. Полезные и достаточные комментарии?	5	6	7	6	3	6	0.5	2	5	7	6	5	3	5.75	0.6875	2	4.5
3. Пропуски строк и пробелы?	2	2	1	4	3	2.25	1.1875	3	5	7	7	4	2	5.75	1.6875	3	4.85
4. Понятна ли логика нижнего уровня?	7	6	7	6	4	6.5	0.25	1	3	6	6	5	3	5	1.5	3	5
5. Схема проектирования верхнего уровня?	7	7	7	6	4	6.75	0.1875	1	6	7	6	6	4	6.25	0.1875	1	6.35
6. Хорош ли алгоритм?	7	7	7	6	4	6.75	0.1875	1	3	6	6	6	3	5.25	1.6875	3	5.7
7. Легко ли понять программу в целом?	7	6	6	6	4	6.25	0.1875	1	7	6	6	6	4	6.25	0.1875	1	5.35
8. Легко модифицировать?	7	7	1	6	3	5.25	6.1875	6	5	7	6	4	2	5.5	1.25	3	4.5
9. Не зависит от компилятора?	6	3	7	7	3	5.75	2.6875	4	1	2	4	5	2	3	2.5	4	3.65
10. Машинно-независима?	6	6	7	7	4	6.5	0.25	1	1	6	3	7	2	4.25	5.6875	6	5.2
11. Были бы Вы горды, написав эту пр-му?	1	6	1	1	3	2.25	4.6875	5	1	7	6	6	3	5	5.5	6	4.5
12. Разумно ли используются структуры данных?	7	6	7	7	4	6.75	0.1875	1	7	7	6	5	3	6.25	0.6875	2	6.65
13. Легко ли отлаживать эту программу?	7	6	7	6	4	6.5	0.25	1	3	6	6	4	2	4.75	1.6875	3	4.8
						Согл.	Среднее	Дисперс	Разброс	Групповое среднее							
1. Хороши ли имена переменных?	3	7	5	7	2	5.5	2.75	4	5.7								
2. Полезные и достаточные комментарии?	2	4	5	1	2	3	2.5	4	4.5								
3. Пропуски строк и пробелы?	6	5	7	7	3	6.25	0.6875	2	4.85								
4. Понятна ли логика нижнего уровня?	6	4	5	7	2	5.5	1.25	3	5								
5. Схема проектирования верхнего уровня?	7	7	6	7	4	6.75	0.1875	1	6.35								
6. Хорош ли алгоритм?	2	5	7	7	2	5.25	4.1875	5	5.7								
7. Легко ли понять программу в целом?	7	4	6	1	2	4.5	5.25	6	5.35								
8. Легко модифицировать?	5	4	6	1	2	4	3.5	5	4.5								
9. Не зависит от компилятора?	1	3	7	1	2	3	6	6	3.65								
10. Машинно-независима?	7	6	7	7	4	6.75	0.1875	1	5.2								
11. Были бы Вы горды, написав эту пр-му?	2	5	6	1	2	3.5	4.25	5	4.5								
12. Разумно ли используются структуры данных?	7	6	7	7	4	6.75	0.1875	1	6.65								
13. Легко ли отлаживать эту программу?	3	5	6	7	2	5.25	2.1875	4	4.8								

Рис.1.Отчет об «оценке равных» от 21.03.1996

На рис.1 приведен реальный отчет о процедуре «оценки равных», проведенной 21.03.1996 г. в рамках учебного процесса в Пермском госуниверситет. Он включает в себя данные об оценке пяти программ по тринадцати субъективным вопросам, средние оценки по каждой программе и групповые средние, ряд вычислимых характеристик для оценки достоверности оценок программы (о них речь пойдет в следующем разделе), информацию о ранжировании рецензируемых программ, ответы на заключительные вопросы о процедуре рецензирования. Студенческая группа из пяти человек на жаргоне

именовалась «звездочкой» в память об октябрьских звездочках. Такие отчеты распечатывались для всех челнов группы. Перед вручением отчета студенту на отчете звездочками отмечались программа, представленная на рецензирование этим студентом, и столбики с оценками, проставленными этим студентом.

Отметим, что групповые средние оценки задают студенту некоторый ориентир: как выглядит его программа на фоне других программ группы.

3. Критерии достоверности полученных оценок: разброс, дисперсия, степень согласованности

Естественный вопрос, который невольно возникает к процедуре «оценки равных», – до какой степени можно доверять полученным оценкам. Для ответа на этот вопрос попробуем проанализировать собранные статистические материалы.

Шнейдерманом приводятся данные по трем экспериментам, в каждом из которых участвовало по 5 человек (профессиональных программистов). Для простоты сопоставления результатов из всех наших экспериментов будем учитывать только те, в которых группы также состояли из пяти человек. В опыте 1994 г. таких групп было три: две из студентов четвертого курса и одна – второго. В опыте 2015 г. таких групп было шесть – четыре из студентов первого курса и две – второго. К сожалению, Шнейдерман не приводит первичных данных по своим экспериментам. Соответственно, и производные показатели для его групп рассчитать не удастся.

Итак, чем можно подтвердить или опровергнуть достоверность полученных оценок? Перове, что приходит в голову, – использовать в качестве меры доверия дисперсию и/или разброс оценок. Если дисперсия и разброс малы, это означает, что все рецензенты поставили близкие оценки, и велика вероятность того, что эти оценки достаточно объективно отражают истинное положение дел. Но что означает ситуация, когда, дисперсия и/или разброс велики? Что достовернее: оценки 2, 3, 4, 5 с разбросом 3 и дисперсией 1.25 или оценки 7, 7, 7, 1 с разбросом 6 и дисперсией 3 (оба примера – реальные данные)?

Опыт показывает, что большой разброс – не исключение, а правило. Так в исследовании 2014-15 гг. в ответах на субъективные вопросы только в 16% случаев разброс был нулевым (т.е. все рецензенты поставили одинаковые оценки). В 30% случаев разброс был максимальным – 6. Ответы с разбросом от 0 до 2 составили 38% процентов, а с разбросом от 5 до 7 – 54%.

С дисперсией дело обстоит несколько лучше: большая дисперсия встречается реже, чем большой разброс. По той же

группе ответов на субъективные вопросы в 33% случаев дисперсия была меньше 1, в 39% случаев – меньше двух, в половине случаев – меньше четырех. при условии, что максимальная дисперсия была равна 12.

Однако использовать дисперсию в качестве меры объективности полученных оценок тоже не удастся. дело в том, что мы имеем дело с очень маленькими группами данных. Малая длина наборов не дает возможности сгладить случайные отклонения то или иной оценки. В результате маленькая дисперсия не всегда может служить показателем более объективной оценки. Например, какие оценки считать более достоверными: (7, 7, 7, 3) – средняя 6, дисперсия 3; (34, 5, 6, 7) – средняя 5.5, дисперсия 1.25; (5, 4, 4, 7) – средняя 5, дисперсия 1.5?

Прежде, чем предложить следующий критерий объективности полученных оценок, попробуем разобраться, с чем связаны неудачи двух предыдущих попыток, чем может быть вызвано различие оценок одной и той же программы. Причин может быть две: субъективизм в оценке программы и различное понимание вопроса разными рецензентами. Какая из причин преобладает в нашем случае? За каждой оценкой стоит некоторое объективное свойство рецензируемой программы. Опыт и уровень подготовки всех рецензентов примерно одинаков. Поэтому можно предположить, что резкий разброс оценок по тому или иному вопросу связан не столько с субъективизмом в оценке свойств программы, сколько с различным пониманием одного и того же вопроса. Такое различие будет выражаться в том, что несколько рецензентов оценят некоторое свойство программы примерно одинаково, а один даст оценку, резко отличающуюся от прочих. Следовательно, нужен показатель, который позволял бы учесть близкие оценки и отбросить резко отличные. В качестве такого показателя предлагается использовать «степень согласованности набора оценок». оценки будем называть согласованными, если разность между ними не превышает единицы. Набор оценок будем называть согласованным, если разность между максимальной и минимальной оценкой в наборе не превышает единицы. Длиной набора будем называть количество оценок в наборе. Согласованностью (степенью

согласованности) набора оценок будем называть длину его максимального согласованного поднабора. Например, для набора оценок (7, 7, 7, 1) согласованность равна трем; для набора (6, 5, 5, 4) – тоже трем; для набора (2, 3, 4, 5) – двум. Степень согласованности набора (1, 3, 5, 7) формально равна единице. Но из психологических соображений ее удобно считать равно нулю. А такой набор назвать полностью рассогласованным.

Если длина наборов одинакова, в качестве степени согласованности можно использовать абсолютную величину. Для сравнения наборов различной длины согласованность удобно указывать как отношение абсолютной величины степени согласованности к длине набора или пересчитывать эту дробь в проценты.

Надо отметить, что степень согласованности не вызывает никаких разночтений в понимании: чем больше – тем лучше.

Рассмотрим ответы на субъективные вопросы с точки зрения согласованности оценок. По данным 2015 г. в 16% случаев мнение всех четырех рецензентов совпало полностью (разброс ответов был равен нулю). В 15% случаев разброс был равен 1. То есть все четыре ответа были согласованы в 31% случаев. Еще в 44% случаев оказались согласованы мнения трех экспертов из четырех возможных. То есть суммарно в 75% случаев по крайней мере три рецензента из четырех поставили оценки, отличающиеся не более, чем на единицу. С другой стороны, абсолютно рассогласованных наборов оказалось чуть более 1%. Средняя степень согласованности оказалась чуть больше трех или 76%.

В 1994 г. картина принципиально не отличалась. Средняя согласованность составила 66.75%. Согласованные ответы не менее, чем трех рецензентов были получены в 58.5% случаев. Совершенно рассогласованных оценок оказалось 1.5%.

По вопросам о процессе рецензирования равными по рангу в 2015 г. средняя согласованность составила 71% (3.55 из 5 возможных). Все 5 оценок были согласованы в 20% случаев, 4 оценки – еще в 30%, 3 оценки – еще в 35%. То есть в половине всех случаев мнения по крайней мере четырех экспертов из пяти отличались не более, чем на единицу. Мнение трех

экспертов из пяти оказалось согласовано в 85% случаев.

В 1994 г. средняя согласованность была 60% (3 из 5). Согласованность в 60% и выше была получена в 75% случаев.

Такая высокая степень согласованности оценок показывает, что рецензирование действительно способно обнаружить некоторые объективные свойства программ.

4. Оценка процедуры рецензирования участниками процесса

Интересные результаты дает анализ ответов на вопросы о процессе рецензирования равными по рангу. Далее в этом разделе «семибальные» ответы свернуты по правилу: «положительный ответ» – 5, 6, 7 баллов, «отрицательный ответ» – 1, 2, 3 балла.

На вопрос «Узнали ли Вы что-либо полезное о стиле программирования?» в 1994 г. положительно ответили 20% младшекурсников, 50% старшекурсников и 67% профессионалов (данные Шнейдермана за 1977-78 гг.), отрицательно – 60% младшекурсников и 30% старшекурсников (по профессионалам данных нет). Похоже, что по мере повышения профессионализма возрастает самокритичность программистов и их готовность воспринять опыт коллег.

Готовы были изменить стиль программирования 60% младшекурсников, 80% старшекурсников и 80% профессионалов. Признавали полезность рецензирования для улучшения стиля программирования в своей организации 100% младшекурсников, 90% старшекурсников и 73% профессионалов.

Таким образом в совокупности студенческие ответы на эти три вопроса звучали примерно так: «Ничего нового для себя я не узнал. Однако при регулярной оценке равных программистов бы иначе. И вообще, оценка равных – вещь чрезвычайно полезная!»

Юношеская склонность к некоторой переоценке своих возможностей очень хорошо видна в ответах на вопросы 1 и 3. Звучат они так: «Самому мне это, конечно, не надо, а вот для других – очень полезно!» По мере роста профессионализма ответы на эти вопросы сближаются.

В 2015 г. в процедуре принимали участие только студенты I и II курса, т.е. младшекурсники. Интересно сравнить их

ответы с ответами 1994 г. Сводная информация собрана в табл.1

Таблица 1. Процент участников анкетирования, давших ответ «да» (5, 6 или 7 по семибальной шкале) на вопросы о процессе рецензирования равными по рангу

Вопрос	1994 г. II курс	1994 г. IV курс	1978 г. Профи	2015 г. I курс	2015 г. II курс
1. Узнали ли Вы в ходе рецензирования что-либо полезное о стиле программирования?	20	50	67	72	63
2. Изменили бы Вы стиль написания Ваших программ, если бы Вам сказали, что рецензирование равными по рангу будет проводиться каждые 6 месяцев?	60	80	80	62	75
3. Считаете ли Вы, что процесс рецензирования равными по рангу может оказаться эффективным средством улучшения программирования в Вашей организации?	100	90	73	74	94

Второкурсники опять показали то же увеличение ответов «да» при переходе от первого вопроса к третьему, что и их предшественники 20 лет назад. Хотя сегодня оно выглядит чуть менее ярко (был прирост в 40% – от 50 до 90, а стал в 30 – от 63 до 94). А вот первокурсники ведут себя гораздо скромнее.

Следующий вопрос – анонимность. По этому поводу наблюдалось изрядное единодушие 20 лет назад и такое же единодушие сегодня. Но смысл этого единодушия изменился. 20 лет назад только 40% участников всех трех категорий ответили на этот вопрос «да», 60% – «нет». В 2015 г. «да» сказали 72% первокурсников и 88% второкурсников. Видимо, переход от работы в аудитории «всем в одно время и в одном месте» к общению через e-mail и индивидуальной работе способствует повышению уровня анонимности.

Последние вопросы нацелены на улучшение процесса «оценки равными». Это общий запрос предложений по совершенствованию процедуры и четыре вопроса по изменению списка вопросов. Цель этих вопросов не только в том, чтобы получить информацию о недостатках процесса и предложения по его улучшению. Еще одна цель состоит в том, чтобы заставить студента не просто применять

рекомендованные критерии качества программы, но и осмысливать их, выявлять их недостатки и искать пути их устранения.

5. Заключение

Накопленный на сегодня опыт показывает что метод «оценки равных»

1) является реально работающим методом экспертной оценки, который обеспечивает проведение коллективной оценки и позволяет сделать это достаточно дешево;

2) может быть реально применим в области computer science

Применение «оценки равных» в области computer science возможно:

1) для оценки качества программ;

2) для оценки квалификации программистов;

3) для обучения программированию.

Возможность оценки качества программ опирается на тот факт что процедура «оценка равных» позволят выявить некоторые объективно присущие рецензируемой программе свойства.

Достоверная оценка квалификации программистов базируется на противопоставлении «оценки равных» и «оценки начальником». Если «оценка равных» способна выявить объективные

свойств программы, она точно так же выявит объективные свойства программиста.

С точки зрения методики обучения программированию «оценка равных» включает в себя следующие положительные моменты:

1. Студенты (прежде всего младшекурсники) получают ориентир, список требований, предъявляемых к программе. Причем ориентир не просто словесный. Они должны сознательно применить эти требования при оценке нескольких программ. Преследуемая при этом методическая цель состоит в том, чтобы довести применение этих критериев до автоматизма при написании собственных программ.

2. Студенты получают еще одну оценку своего программистского уровня, причем не от преподавателя, а от своих коллег. Наличие нескольких рецензий повышает объективность оценки.

3. Происходит обмен опытом между студентами.

4. Студенты осваивают один из приемов совместной работы.

5. Совместная работа способствует преодолению через чур личностного отношения к программе, отождествления студентом себя со своей программой. Причем делается это в достаточно мягкой форме (каждая программа обсуждается совместно с другими программами, без указания имени автора и имен рецензентов).

Все это позволяет рекомендовать процедуру «оценки равных» для систематического использования в учебном процессе.

Благодарности. Автор благодарит студентов факультета бизнес-информатики Пермского филиала НИУ ВШЭ А.Д. Кучева и К.Э. Фалетова за помощь в сборе и подготовке информации.

Литература

1. *Дрогобыцкий И.Н.* Системный анализ в экономике: учеб. пособие /И.Н. Дрогобыцкий. – М.: Финансы и статистика; ИНФРА-М, 2009.
2. *Плаксин М.А.* «Оценка равных» как прием совместной работы программистов и обучения программированию. Опыт экспериментального исследования. //Вестник пермского университета. Научный журнал. Математика. Механика. Информатика. Выпуск 1. 1997, с.222-231.
3. *Шнейдерман Б.* Психология программирования. Человеческие факторы в вычислительных и информационных системах /Шнейдерман Б. – М.: Радио и связь, 1984 г.