

УДК 528.8.04, 528.88

Система автоматизированной оценки заданий по программированию NSUts и ее применение в учебном процессе

Иртегов Д.В., доцент, руководитель лаборатории НГУ

тел.: +79139877421; e-mail: fat@nsu.ru

Нестеренко Т.В., ст. преподаватель НГУ, н.с. ИСИ

тел.: +73833307068; e-mail: nest@iis.nsk.su

Чурина Т.Г., к.ф.-м.н., доцент НГУ, с.н.с. ИСИ

тел.: +73833306253; e-mail: tanch@iis.nsk.su

ФГАОУ ВО «Новосибирский национальный исследовательский государственный университет»

Факультет информационных технологий

Институт систем информатики им. А.П. Ершова СО РАН,

Система автоматизированной оценки NSUts разработана в НГУ. Эта система и ее предшественник chjudge используются в учебном процессе и в спортивном программировании с 2000 года. В статье обсуждаются архитектура системы и ее возможные применения для задач за пределами учебного процесса.

Ключевые слова. Компьютерное тестирование, автоматическая оценка, обучение программированию, спортивное программирование, олимпиады по программированию.

Интенсивное развитие информационных технологий диктует необходимость постоянного совершенствования структуры и содержания образования в области высоких технологий. Конкурентоспособность отраслей хозяйства России во многом зависит от уровня используемых информационных технологий и от уровня квалификации специалистов по информатике. Проблема подготовки квалифицированных кадров по информатике и информационным технологиям имеет особую значимость. Повышенные требования к уровню профессионализма выпускников вузов предъявляют и научные школы институтов РАН, и бурно растущая область IT-индустрии, решающая все более сложные практические задачи.

Решение этой задачи может достигаться за счет расширения подготовки IT-специалистов, повышения качества этой подготовки и снижения стоимости такой подготовки, если это не сопровождается потерей качества.

Популярным методом повышения качества образования и снижения его себестоимости без потери качества является применение автоматизированного тестирования и оценки знаний и навыков. Этот метод в различных формах применяется в школьном, в высшем и в послевузовском обучении. В частности, многие сертификаты, например, Microsoft Certified System Engineer, Cisco Engineer, Certified Lotus Professional выдаются на основе автоматизированного тестирования.

По сравнению с обычными формами контроля знаний и навыков, такими как устные экзамены, контрольные работы, курсовые и дипломные проекты, автоматизированное тестирование и оценка требуют меньших трудозатрат. Такие средства обеспечивают гораздо более плотный контроль качества обучения, чем это было бы возможно в рамках традиционных форм. Они удачно дополняют традиционные формы контроля и могут использоваться для повышения качества образования при незначительном росте его себестоимости.

Программирование представляется очень удачной дисциплиной для автоматической оценки. Эффективным комплексным средством проверки знаний и

навыков программиста является написание программы, соответствующей заданным требованиям, и тестирование полученной программы.

Разработка автоматизированной системы оценки учебных и тестовых заданий по программированию, имеющей эффективную техническую защиту от мошенничества и способной работать под присмотром специалиста средней квалификации, например, школьного учителя информатики, и разработка методик тестирования различных навыков, связанных с программированием (основы программирования, алгоритмы и структуры данных, системное программирование, работа с базами данных, математическое моделирование, олимпиадные задачи), могли бы решить целый ряд задач, связанных с совершенствованием качества образования в области информатики и привлечения и закрепления молодежи в сферы науки, образования и высоких технологий. Такая система могла бы использоваться для:

1. промежуточного тестирования знаний и навыков студентов высших и среднетехнических учебных заведений;
2. спортивного программирования;
3. профориентационного тестирования;
4. тестирования в службе занятости и кадровых агентствах при отборе на вакансии, требующие навыков программирования;
5. одного из этапов собеседования в IT-компаниях;
6. послевузовского, в том числе дистанционного, образования.

Первая в НГУ система для автоматической оценки заданий по программированию, chjudge, была создана под руководством авторов студентом ФИТ НГУ Е. А. Четвертаковым в 2002 году. На основе опыта ее эксплуатации были выработаны требования к более совершенной системе [1]. В 2008 году, используя исходные коды системы chjudge и учитывая сформулированные требования, была начата разработка системы, получившей название NSUts. Накоплен существенный опыт применения NSUts в учебном процессе и для спортивного программирования.

NSUts используется для проведения Открытой Всесибирской олимпиады по информатике и программированию имени И. В. Поттосина, и на школьных и студенческих олимпиадах [2]. Система также используется для тренировок сборной НГУ по программированию.

Практика показывает, что олимпиады по информатике и программированию являются источниками качественных и интересных задач, которые полезно использовать в дальнейшем при обучении

Важной частью использования системы NSUts является ее применение в промежуточном контроле студентов первого курса при изучении курса «Программирование». Лектор и преподаватели составляют задачи, подготавливают тесты и выставляют баллы за каждый тест. Для проверки умения применять изученные алгоритмы студентам предлагается решать, в том числе и задачи олимпиадного характера.

В системе преподаватель может создать свой тур, настроить время его начала и окончания, создать нужное количество задач, загрузить тесты к задачам, при необходимости выложить теоретический материал, используемый при решении задач тура. Тесты составляются таким образом, чтобы выявлять основные ошибки, которые делаются при реализации алгоритмов. Для выявления неэффективных решений преподаватель может настроить временные ограничения на выполнение программы на каждом тесте. Также во время сдачи задач и после преподаватель может перетестировать любое решение, посмотреть ответы, даваемые выбранным решением, посмотреть статистику сдачи решений по каждой задаче и статистику прохождения тестов.

На занятиях студент решает задачи за компьютером и посылает свои решения на проверку в систему NSUts. Он может контролировать, на каких тестах его программа дает неверный ответ, и попытаться исправить ошибки. Преподаватель видит результаты проверок, может посмотреть текст программы для анализа используемого алгоритма, оценки качества написанного кода. Такой процесс контроля является хорошей предварительной подготовкой к сдаче лабораторных заданий.

В течение последних четырех лет в ряде групп прием лабораторных работ проводится в первом семестре полностью посредством системы NSUts. Каждое занятие на семинарах подкрепляется заданием из 3-6 задач на неделю, которые нужно сдать в автоматизированную систему NSUts. Эти задачи студентам не только нужно написать, их решения должны пройти все тесты, а затем необходимо защитить это решение перед преподавателем на занятии в компьютерном классе. Разработаны критерии оценки задач, которые помогают стимулировать самостоятельную внеаудиторную работу студентов.

Задачи, входящие в наборы заданий на каждую неделю, с одной стороны, помогают закрепить тот материал по изучению языка программирования, который разбирается на семинарах, с другой, реализовать алгоритмы, представленные в курсе лекций. Работа в системе NSUts дисциплинирует обучающихся, заставляет их соблюдать правила работы с входными и выходными данными, продумывать все тонкости решения, не забывать про граничные условия и про эффективность алгоритмов.

Система состоит из сервера, базы данных, тестирующих клиентов и клиентского программного обеспечения (веб-браузеров).

Сервер реализует аутентификацию и авторизацию доступа со стороны пользователей и большую часть бизнес-логики системы, а также предоставляет веб-интерфейс для пользователей и веб-сервис для тестирующих клиентов.

База данных отвечает за транзакционное хранение данных системы, и отслеживает часть правил бизнес-логики, главным образом за счет ограничений целостности данных (SQL constraints).

Тестирующие клиенты осуществляют компиляцию решений, отправленных участниками, и запускают откомпилированный код на заранее подготовленных тестовых данных. Для этого используются стандартные компиляторы и стандартные или минимально модифицированные среды исполнения. Так, если в правилах тестирования указан язык MinGW C/C++ 4.6.1, на клиенте должны быть развернуты среда исполнения MinGW и соответствующие компиляторы.

Данный подход противоположен подходу, используемому в некоторых системах автоматической оценки, например, EduCAD [3], где система реализует собственный компилятор или интерпретатор и среду исполнения языка программирования. Использование стандартных компиляторов и сред исполнения обеспечивает ряд важных преимуществ.

Компилятор и среда исполнения языка программирования – это сложные программные комплексы. При их самостоятельной разработке, у разработчика гораздо меньше ресурсов для их доводки и поддержки, чем у создателей популярной коммерческой или разрабатываемой сообществом среды разработки. В результате, может оказаться, что тестируемые будут проверяться не на знание языка и навыков программирования, а на навыки обхода ошибок в тестирующей среде.

Во-вторых, использование стандартных сред разработки позволяет легко добавлять поддержку новых компиляторов и даже языков. Это особенно важно, когда учебный план или правила соревнований требуют проверки заданий на каком-то определенном языке и диалекте. По состоянию на 10 июня 2016 года, NSUts

поддерживает 28 языков (считая реализации C/C++ в Visual Studio 2015 и MinGW 4.6.1 как разные языки).

Еще одной важной особенностью NSUTs, отличающей ее от других систем, например, ejudge [4] и Kattis [5], является проверка заданий на выделенном компьютере. Регламент многих соревнований и правила приема учебных заданий подразумевают ограничение времени исполнения решений. Как правило, это интерпретируется как ограничение значений счетчика ОС «время центрального процессора».

Но указанный счетчик учитывает не только время вычислений, но и время переключений контекста процесса, промахи кэша, конкуренцию за циклы доступа ОЗУ с другими процессорными ядрами. Проведенные измерения [6] показывают, что это может приводить к отклонениям значений этого счетчика в многозадачной среде на десятки процентов от значений при работе той же программы на выделенной машине. Под многозадачной средой подразумевается практически любая ситуация, когда тестовый клиент используется для чего-то еще, кроме тестирования – например, когда сервер системы и тестируемое задание работают на одной машине, как это сделано в ejudge.

Вопрос о том, что же в многозадачной среде означает вердикт «превышен лимит времени», оказывается неразрешимым ни практически, ни теоретически. Исполнение решений на выделенных машинах, хотя и усложняет развертывание системы, но позволяет исключить этот вопрос из обсуждения, что облегчает подготовку тестов и снижает количество обоснованных апелляций.

Кроме того, выделенные тестовые клиенты увеличивают масштабируемость системы. Во время крупных соревнований с 300-500 участниками мы задействуем 10-16 компьютеров для параллельного тестирования.

Использование системы NSUTs в учебном процессе и для самостоятельной проверки своих знаний в процессе дистанционного обучения способствует совершенствованию профессиональных, аналитических, системных и коммуникационных компетенций бакалавров и магистров, аспирантов, молодых учёных, профессорско-преподавательского состава.

Список литературы

1. Боженкова Е.Н., Иртегов Д.В., Киров А.В., Нестеренко Т.В., Чурина Т.Г. Автоматизированная система тестирования NSUTs: Требования и разработка прототипа // Вестник НГУ. Сер. Информационные технологии. – 2010. – Том 8; № 4. – С. 46–53.
2. Открытая Всесибирская олимпиада по программированию им. И. В. Поттосина [Электронный ресурс]. URL: <http://olympic.nsu.ru/>
3. Веретенников М.В. Автоматизация проверки компьютерных программ в технических дисциплинах. // Дистанционные образовательные технологии. Выпуск 1: Пути реализации. Сборник научных трудов. Томск: ТУ СУР, 2004. –С. 52-61.
4. Ejudge [Электронный ресурс, режим доступа – свободный]. URL: <https://ejudge.ru> (10.06.2016)
5. E. Enstrom, G. Kreitz, F. Niemela, P. Soderman, V. Kann, Five Years with Kattis – Using an Automated Assessment System in Teaching, Proceedings of the 2011 Frontiers in Education Conference, p.T3J-1-1-T3J-6, October 12-15, 2011
6. Свиридов В.С., Измерение и контроль потребления ресурсов программами на машинах с многоядерными процессорами, выпускная квалификационная работа бакалавра, НГУ, 2015