

АВТОМАТИЗИРОВАННОЕ ТЕСТИРОВАНИЕ ЛАБОРАТОРНЫХ РАБОТ ПО СВЯЗАННЫМ С ПРОГРАММИРОВАНИЕМ ДИСЦИПЛИНАМ ПРИ ДИСТАНЦИОННОМ ОБУЧЕНИИ

И.В. Некрашевич¹, Л.А. Глухова²

¹ *Белорусский государственный университет информатики и радиоэлектроники, Минск, Беларусь, archimedis@tut.by*

² *Белорусский государственный университет информатики и радиоэлектроники, Минск, Беларусь, glukhova@bsuir.by*

Abstract. This article describes the possibility of using Continuous Integration tools for education. The main advantage of that process for students is the possibility to check their solutions in fully automated environment. This would also develop some additional skills which are very useful in real software developing company.

Большая часть технических специальностей вузов предусматривает обучение студентов связанным с программированием дисциплинам. Важной особенностью проверки компьютерных программ является то, что сами тексты созданных программ проверять достаточно сложно, а иногда проверка текста программы вообще невозможна.

При очной форме обучения преподаватель, проверяющий прикладные компьютерные программы, может попросить студента объяснить соответствие алгоритма работы его программы исходному тексту и, таким образом, несколько облегчить процесс проверки. При дистанционной или заочной форме обучения преподаватель, проверяющий подобную прикладную программу, часто не имеет возможности услышать комментарии студента по алгоритму ее выполнения и проверяет ее только по результатам работы программы.

Выходом из данной ситуации могло бы стать тестирование результатов работы программы с возможным последующим анализом кода программы. При использовании небольшого набора правил работы со входными и выходными данными процесс проверки результатов работы компьютерных программ можно автоматизировать. Так как произвести полный анализ исходного кода в автоматическом режиме не представляется возможным, то имеет смысл производить анализ кода на наличие типичных ошибок и опечаток. Таким образом, возникает проблема автоматизированного тестирования программ.

В последнее время системы автоматизированного тестирования программ очень часто используются для проверки учебных и олимпиадных задач по информатике. Например, система проверки программ используется при проведении международных студенческих командных олимпиад под эгидой Association for Computing Machinery (ACM) [1]. Принцип работы таких программ состоит в том, что студенты отправляют свои решения на сервер, где они проверяются системой. Процедура проверки заключается в том, что на вход тестируемой программы поочередно подаются тестовые наборы данных, при обработке каждого из которых программа генерирует выходные данные. Проверяющая система сравнивает выходные данные программы с эталонными и на основе этого делает заключение о корректности предоставленного решения.

Для подготовки команд на студенческие олимпиады используются несколько другие системы, например: «EJudge», «Олимпия», система автоматической проверки решений задач по программированию [2].

На производстве для осуществления процесса автоматического контроля качества исходного кода используется непрерывная интеграция, реализованная при помощи

целого набора приложений, тесно взаимодействующих между собой. Так как большинство из этих программ бесплатны и находятся в свободном распространении, то целесообразно использовать их для систем дистанционного обучения.

На рисунке 1 представлена модель тестирования лабораторных работ по связанным с программированием дисциплинам. Модель базируется на использовании следующих компонентов:

1) система управления версиями – система хранения исходного кода, куда студенты заносят результаты выполнения своих лабораторных работ;

2) сервер непрерывной интеграции – система, отвечающая за запуск процесса тестирования и генерацию отчётов;

3) модуль сборки и статического анализа – модуль управления компиляцией и статическим анализом исходного кода;

4) компилятор языка программирования;

5) статический анализатор – система проверки исходного кода на выявление мест, возможно содержащих ошибки;

6) модуль тестирования – модуль, отвечающий за проведения квалификационных тестов;

7) модуль управления почтой – модуль, производящий информационную рассылку информации о результатах тестирования.

Использование предлагаемой модели тестирования лабораторных работ позволит повысить качество обучения студентов дистанционной формы по связанным с программированием дисциплинам.



Рисунок 1 – Схема автоматизированной проверки лабораторных работ

Литература

1. Васильев, В.Н. Командный чемпионат мира по программированию АСМ 1998/1999. / Васильев В.Н., Парфенов В.Г. // Спб: СПбГИТМО(ТУ), 1998. – С. 112.

2. Цымблер, М.Л. Система автоматической проверки решений задач по программированию, использующая сетевые технологии/ М.Л. Цымблер, М.М. Арсламбеков // Телематика'99: Тез. докл. Всеросс. науч.-метод. конф. (8–11 июня 1998 г., Санкт-Петербург). СПб: Вузтелекомцентр. 1999. – С. 143.