

УДК 519.688, 519.682.3, 004.42
AMS MSC2020: 68W40, 68Q25, 97N80

Автоматическое тестирование студенческих программ

Новиков М. Д.

Московский государственный университет им. М. В. Ломоносова,
факультет вычислительной математики и кибернетики

Аннотация. В статье описываются две системы, предназначенные для тестирования программ, выполняемых студентами первого курса факультета ВМК МГУ в рамках практикума на ЭВМ. Системы разрабатываются на факультете ВМК с 2017 года. Первая система тестирует программы на языке Паскаль, а вторая — на языке Ассемблера.

Ключевые слова: язык Паскаль, язык Ассемблера, тестирующие программы, программирование, информатика.

Введение

Языки программирования Паскаль и Ассемблер изучаются студентами факультета ВМК МГУ на первом курсе в течение многих лет; задачи для практических занятий по Паскалю берутся, в основном, из книг [9] и [8], а задачи по Ассемблеру берутся из книг [1] и [2]. В этих учебных пособиях собрано много задач на различные темы; они охватывают все основные конструкции языков Паскаль и Ассемблер. Возникает проблема тестирования написанных студентами программ, то есть проверки правильности их работы на различных наборах исходных данных. Проверка простых программ обычно не вызывает затруднений, а проверка сложных программ предполагает ввод с клавиатуры нетривиальных исходных данных и сверка результатов, выдаваемых студенческой программой, с правильными результатами. Это занимает много времени и часто не гарантирует выявление всех ошибок. Указанные сложности тестирования программ побудили автора к созданию систем, позволяющих

производить тестирование в автоматическом режиме, то есть запускать тестируемую программу для различных, заранее заготовленных тестовых данных и сравнивать результаты, выдаваемые программой, с эталонными (правильными) результатами.

1. Требования к студенческим программам

В формулировках задач из [1, 2, 8, 9] указаны требования, предъявляемые к той или иной студенческой программе. К некоторым задачам надо написать полную программу на Паскале или Ассемблере; алгоритмы для решения других задач должны быть оформлены в виде фрагмента программы, процедуры или функции. Для некоторых задач из [1, 2, 8, 9] дополнительно указаны требования, предъявляемые к методам их решения: например, необходимо описать подпрограмму с определенным набором параметров, реализовать рекурсивный алгоритм, использовать конкретный тип данных языка Паскаль или Ассемблера и т. д. Целью таких требований является обучение студентов не только составлять программы, но и осваивать различные приемы программирования и конструкции изучаемых языков программирования.

2. Описание систем автоматического тестирования программ

Возможности обеих систем следующие.

- 1) Можно тестировать либо полную программу, либо фрагмент программы. Если (по условию задачи) необходимо написать именно фрагмент программы, то система тестирования добавляет к ней блок, дополняющий ее до полной программы, которая затем тестируется. В системе тестирования программ на Паскале есть также возможность тестировать полную программу наряду с автономным тестированием входящей в нее процедуры или функции. Это необходимо, если по условию задачи требуется составить полную программу и описать внутри нее подпрограмму заданного вида.
- 2) Проверяется выполнение требований к алгоритмам, указанным в [1, 2, 8, 9]:

- а) запрет или, наоборот, обязательное использование каких-либо конструкций языка Паскаль или Ассемблера;
- б) реализация заданного конкретного алгоритма решения задачи.

Для проверки выполнения требований к алгоритмам производится анализ тестируемой программы. При этом подсчитывается количество вхождений определенных слов в программу, например, while, if, procedure и др. По результатам анализа делается заключение о соответствии программы указанным требованиям.

3) Проверяется корректность выполнения тестируемых программ:

- а) в задачах на использование внешних файлов (Паскаль) проверяется закрытие их в программе по окончании работы с ними; такая ошибка обычно не фиксируется операционной системой;
- б) в задачах на динамические структуры данных (Паскаль) проверяется корректность использования динамической памяти; для этого подсчитывается количество обращений к процедурам New и Dispose в процессе выполнения программы;
- в) в задачах на составление процедур и функций (Паскаль) проверяется отсутствие в теле подпрограммы глобальных переменных и способ вызова фактических параметров — по значению или по ссылке;
- г) в задачах на составление программ обработки символьных данных (Ассемблер) проверяется установка флага направления просмотра строк.

Для проверки корректности выполнения тестируемых программ к ним добавляются специальные операторы: например, чтобы проверить закрытие файла в процедуре, система тестирования вставляет операторы close(f); I:=ioreturn;if I=0 then ... (файл не был закрыт) после обращения к этой процедуре.

4) Реализованы также некоторые дополнительные проверки тестируемых программ, в частности:

- а) в задачах, допускающих различные формы вывода результата, любой допустимый вывод считается правильным;
- б) проверяется количество используемых циклов и досрочные выходы из циклов (отсутствие избыточных вычислений) там, где это необходимо по условию задачи;
- в) в задачах на использование ограниченного типа данных (Паскаль) в качестве тестов предлагаются числа, выходящие за допустимый диапазон значений. Возникающая при этом ошибка периода выполнения считается верным ответом.

Каждая тестируемая программа выполняется на заданных наборах исходных данных. По окончании тестирования выдается результат — исходные данные, выданные программой ответы, правильные ответы, общее количество правильных ответов и количество ошибок с указанием типа каждой ошибки.

Обе системы автоматического тестирования написаны в среде Delphi и могут выполняться в ОС Windows; для трансляции тестируемых программ используются компиляторы Free-Pascal и Masm 6.14. Системы могут быть установлены на локальном компьютере и не требуют доступа в Интернет. К настоящему времени составлены тесты примерно к 550 задачам по Паскалю и к 300 задачам по Ассемблеру. Планируется составить тесты к задачам из [1, 2, 8, 9], еще не вошедшим в системы тестирования и реализовать тестирование полной программы на языке Ассемблера наряду с автономным тестированием входящей в нее процедуры.

Системы автоматического тестирования были представлены на двух конференциях: на конференции, посвященной памяти Н. П. Трифонова [5] и на конференции Ломоносовские чтения [6]. По теме опубликованы три статьи [3, 4, 7].

Заключение

Использование систем показало их эффективность. Раньше для проверки студенческой программы преподавателю приходилось вручную вводить сложные исходные данные и сверять результат с пра-

вильным; при этом часто обнаруживались ошибки. Теперь же студенты отлаживают свои программы самостоятельно, принося на проверку уже готовые варианты.

Список литературы

- [1] *Бордаченкова, Е. А.* Задания практикума. 1 курс / Е. А. Бордаченкова, А. А. Панферов. — М. : МАКС Пресс, 2016. — 48 с.
- [2] *Бордаченкова, Е. А.* Задачи и упражнения по языку Ассемблера MASM. — М. : МАКС Пресс, 2020. — 92 с.
- [3] *Новиков, М. Д.* Система автоматического тестирования программ, написанных на языке Паскаль // Альманах современной науки и образования. — 2017. — №6. — С. 68–71.
- [4] *Новиков, М. Д.* Автоматизированный практикум по языку программирования Паскаль // Наука России: цели и задачи. Сборник научных трудов по материалам XVII международной научно-практической конференции 10 октября 2019 г. Часть 1 Изд. НИЦ «Л-Журнал», 2019. — С. 26–31.
- [5] *Новиков, М. Д.* Автоматическое тестирование программ на языке Паскаль // Программирование и вычислительная математика. Тезисы докладов конференции памяти Н. П. Трифонова / под ред. С. А. Абрамова, А. В. Столярова. — М. : МАКС Пресс, 2020. — С. 34–36.
- [6] *Новиков, М. Д.* Система автоматического тестирования студенческих программ на языке Ассемблера // Ломоносовские чтения. Тезисы докладов. Секция вычислительной математики и кибернетики. — М. : Изд-во Московского ун-та, 2021. — С. 116–117.
- [7] *Новиков, М. Д.* Тестирование программ на языке Паскаль, использующих динамическую память // Eastern-European Scientific Journal. — 2021. — Т. 1, №3 (67). — С. 45–47.
- [8] *Пильщиков, В. Н.* Задания практикума на ЭВМ. 1 курс / В. Н. Пильщиков, Н. П. Трифонов. — М. : Издательский отдел факультета ВМК МГУ, 2001. — 34 с.
- [9] *Пильщиков, В. Н.* Язык Паскаль: упражнения и задачи. — М. : Научный мир, 2003. — 224 с.

Библиографическая ссылка

Новиков, М. Д. Автоматическое тестирование студенческих программ // Всероссийская научная конференция «Математические основы информатики и информационно-коммуникационных систем». Сборник трудов. — Тверь : ТвГУ, 2021. — С. 235–240.
<https://doi.org/10.26456/mfcsics-21-32>

Сведения об авторах

Михаил Дмитриевич Новиков
Московский государственный университет им. М. В. Ломоносова, факультет вычислительной математики и кибернетики. Старший научный сотрудник

Россия, 119991 ГСП-1 Москва, Ленинские горы, МГУ им. М. В. Ломоносова, д.1, стр. 52, ф-т ВМК
E-mail: novikov_57@mail.ru