

## Система тестирования задач программирования «Данные – Результат – Связь»

*Зенько Иван Александрович*

УО «Брестский государственный университет им. А.С. Пушкина»

Научный руководитель: доцент кафедры прикладной математики и информатики УО "Брестский государственный университет имени А.С. Пушкина", доцент *Силаев Н.В.*

Предлагаемый нами вариант тестирования программного продукта касается, в первую очередь, тех случаев, когда решение задачи приводит к «множественным» результатам, т.е. несколько различных между собой решений являются правильными по отдельности по отношению к входным данным. Такая ситуация, в частности, нередко наблюдается при решении геометрических задач на построение.

Заметим, что в подобном случае ни тестирование типа «сравнение файлов», ни тестирование типа «сравнение результатов решений» либо не подходят полностью, либо приводят к построению сложных систем сравнения.

Для разрешения проблемы подобного рода мы предлагаем использовать проверку по «контрольным свойствам». Такого рода проверки иногда называют «чеккер-контролем».

В качестве «контрольных свойств» могут выступать специфические для конкретной задачи случаи взаимосвязи исходных данных и результатов решения задачи. Выделение подобных свойств для каждой отдельной задачи и составляет суть написания «чеккер-кода». Обратим внимание лишь на то обстоятельство, что «чеккер-код» может быть оформлен в двух вариантах:

- в виде набора методов (подпрограмм) в составе модуля (библиотеки классов);
- в виде внешнего решения, передаваемого в тестирующую систему как функциональный параметр (делегат).

Общий алгоритм дальнейших действий представлен на функциональной схеме рисунка 1.

Согласно этой схеме «внешними» или «входными» данными рассматриваемой системы тестирования являются:

- набор файлов с исходными данными (только входными). Кстати, их можно будет формировать и по мере тестирования;
- библиотека «чеккер-кодов» проверки связи входных и выходных файлов тестируемой программы;
- сама тестируемая программа в виде исходного или исполняемого кода.

Далее, следуя функциональной схеме, организуется цикл проверок, включающий такие этапы, как

- извлечение из набора файлов с исходными данными очередного файла;
- автоматический запуск на выполнение тестируемой программы, которая в процессе своей работы должна ввести исходную информацию из выбранного на предыдущем этапе файла, имеющего стандартный системный идентификатор;

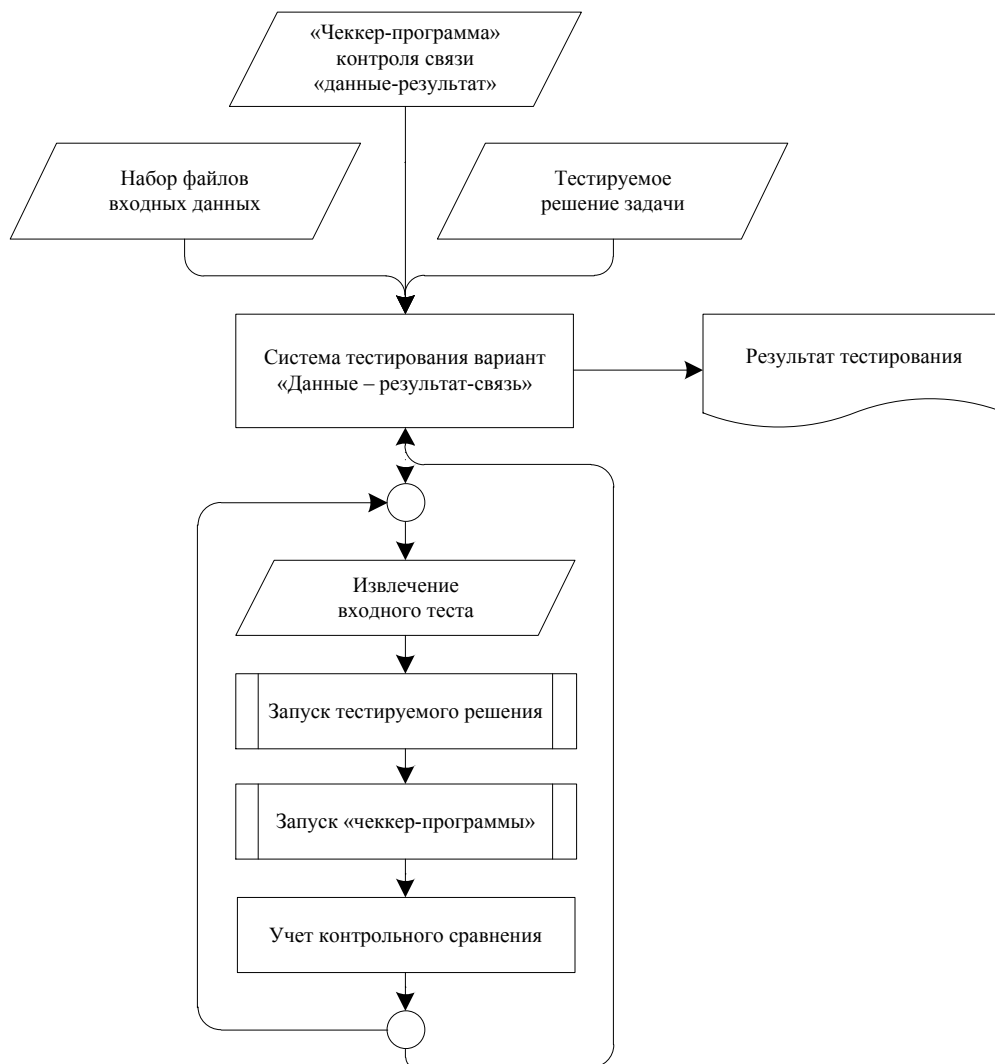


Рисунок 1. – Общий алгоритм.

- автоматический запуск на выполнение «чеккер-программы» (метода, функции) с указанием в качестве ее фактических параметров стандартных имен двух файлов: файла с исходными данными и выходного файла, полученного на предыдущем этапе работы системы в результате работы тестируемой программы. Заметим, что чаще всего «чеккер-программа» представляет собой логическую функцию с результатами работы **True** – тест пройден корректно и **False** – тест не пройден. Возможны и другие варианты: тест пройден частично или степень прохождения теста (в этом случае «чеккер-программа» должна выдавать числовой результат);
- заключительный этап одного шага итераций – учет результатов работы «чеккер-программы» в режиме накопления информации.

К положительным моментам данной системы тестирования по отношению к другим системам можно отнести:

- бóльшую универсальность;
- уменьшение объема входных данных в два раза;
- иногда бóльшую простоту организации «чеккер-программ».

Вместе с этим, при таком подходе надо создавать «чеккер-программы» как третий вид входной информации.