

▪ Модель ИТ-специалиста. Предложена субъективная оценочная модель навыков, способностей и знаний сотрудника. Она определяется на основе резюме, собеседования, полученного образования, накопленного опыта, психологического тестирования и т.д., которые представляются в виде определенных значений коэффициентов.

▪ Модель проекта изделия. Она определяется в виде совокупности следующих компонентов: вида проекта, назначения проекта, сроков разработки, требуемых ресурсов и других компонентов.

В настоящее время разработан программный комплекс (ПК), состоящий из базы данных (БД) для хранения рассмотренных моделей, а также набор приложений, обеспечивающих работу с ними. Ведется накопления реальных данных в БД и адаптация предложенных моделей к реальным условиям производства ИТ-изделий.

На основе предложенного подхода и созданного ПК разрабатывается система поддержки принятия решений при управлении ИТ-проектами, которая позволит уменьшить следующие риски: при открытии новых ИТ-проектов, при подборе сотрудников для их реализации, при планировании производства ИТ-изделий и другие.

**О.А. Васильев (УО «ГГТУ им. П.О. Сухого», Гомель)**  
Науч. рук. **Е.Г. Стародубцев**, канд. физ.-мат. наук, доцент

## **АНАЛИЗ РЕЗУЛЬТАТОВ ТЕСТИРОВАНИЯ С ПРИМЕНЕНИЕМ ГРАФОВ**

Автоматизация анализа результатов тестирования позволяет минимизировать временные затраты на анализ проблем. Автоматизацию можно ввести путем разработки унифицированного формата тестовых сообщений и ошибок, к которым применяются средства анализа. В качестве основного средства для анализа ошибок тестов предлагается использовать систему, построенную на основе графов. Используется взвешенный граф [1], множество вершин которого представлено тестами, с имеющимися стеками вызовов и сообщениями об ошибках, данный граф не может иметь циклов.

Алгоритм можно описать следующим образом. В начале цикла имеется нуль-граф, для начала анализа выбирается произвольная вершина, однако предпочтительно выбирать первую вершину, т.к. для хранения графа используется списковая структура. Начиная с выбранной вершины, идет попарное сравнение с остальными вершинами графа по определенным правилам, которые определяют степень схожести проблем.

Функция, выполняющая роль правила, возвращает числовое значение, описывающее степень схожести проблем. При удовлетворении минимального барьера схожести между вершинами графа создается дуга с весом, равным значению вычисленной функции. Все найденные схожие вершины маркируются, и происходит переход к следующей немаркированной вершине. Цикл повторяется, пока все вершины не будут промаркированы.

На выходе будет получен граф, который представляет собой множество подграфов, отражающих найденные проблемы. Алгоритм предлагает высокую скорость обработки больших объемов данных, т.к. при каждой итерации скорость будет возрастать с исключением вершин. Граф может использоваться как для получения групп проблем по тестам, так и для дальнейшей визуализации, оценки точности сравнения и дальнейшей корректировки правил.

### **Литература**

1. Graph (mathematics) [Электронный ресурс] / Wikipedia. – 2005. – Режим доступа: [http://en.wikipedia.org/wiki/Graph\\_\(mathematics\)](http://en.wikipedia.org/wiki/Graph_(mathematics)). – Дата доступа: 6.03.2014.

**О.А. Васильев** (УО «ГГТУ им. П.О. Сухого», Гомель)  
Науч. рук. **Е.Г. Стародубцев**, канд. физ.-мат. наук, доцент

### **ВИЗУАЛИЗАЦИЯ ПРОЦЕССА ВЫПОЛНЕНИЯ АВТОМАТИЧЕСКИХ ПАРАЛЛЕЛЬНЫХ ТЕСТОВ**

Распараллеливание автоматических тестов имеет как преимущества (увеличение быстродействия, возможность тестирования системы на единовременный доступ), так и недостатки (сложность поддержки подобных тестов и исправления ошибок). Один из способов решения данной проблемы – графическая визуализация программных потоков. В качестве удобного средства отображения параллельного выполнения тестов можно использовать график с временной линией. Вдоль ординат такого графика размещается временная линия с разметкой по часам и минутам; вдоль оси абсцисс – номера потоков, в которых запускаются тесты; на плоскости графика – названия тестов с отметками времени начала и конца.

В качестве платформы для реализации был выбран язык JavaScript (уже существуют реализации похожих графиков, но большинство из них поставляется в виде платных программных комплексов либо имеет