

**Н. А. Иняхин**  
Науч. рук. **Н. Б. Осипенко**,  
канд. физ.-мат. наук, доцент

## АВТОМАТИЗАЦИЯ РЕКОМЕНДАЦИЙ ПО ВЫБОРУ ОПТИМАЛЬНОГО БАНКОВСКОГО ВКЛАДА

Для решения задачи автоматизации рекомендаций по выбору банковского вклада используется многокритериальная оптимизация с учетом мнения трёх экспертов, метод Ранга определения значимости критериев, отбор бесперспективных альтернатив с помощью множества Парето-оптимальных решений и определение лучшей альтернативы методом анализа иерархий.

В качестве исходных данных используется реальная информация о банковских вкладах. Каждый из экспертов оценивает банковские вклады по пятибалльной шкале. Полученные экспертные балльные оценки критериев используются в методе Ранга, в котором наиболее предпочтительным является критерий с максимальным весом. После этого используется выбор множества Парето, он представляет собой отбор перспективных альтернатив, из которых затем отбирается одна лучшая альтернатива. Имеющиеся альтернативы попарно сравниваются друг с другом по всем критериям. Если при сравнении каких-либо альтернатив оказывается, что одна из них не лучше другой ни по одному из критериев, то ее можно исключить из рассмотрения. Для завершения многокритериальной оптимизации используется метод анализа иерархий.

На основании описанной выше схемы выбора оптимального банковского вклада разработано оконное *Windows*-приложение на языке *C#*, в состав которого входят три *Windows*-формы и девять классов. Каждый из классов имеет свой набор полей и методов.

Рассмотрим основные классы. Класс *SortByRate* реализует обобщенный интерфейс *Comparer<Vklad>* и используется для метода сортировки по стоимости. Класс *SortByValuation* реализует обобщенный интерфейс *Comparer<Vklad>* и используется для метода сортировки по окончательной оценке вклада. Класс *Optimization* служит для реализации однокритериальной и многокритериальной оптимизации. Класс *Pareto* – это класс, реализующий выполнение принципа Парето. Класс *Rang* – это класс для реализации метода принципа Парето. Присутствует проверка оценок на согласованность. Класс *AnalysisHierarchies* – класс, реализующий метод многокритериальной оптимизации. Для хранения информации использована СУБД *MySQL*.

Приложение прошло верификацию на тестовых данных. Тематика работы актуальна как для организаций, так и для частных лиц.

**Р. А. Кацора**  
Науч. рук. **Г. Л. Карасёва**,  
канд. физ.-мат. наук, доцент

## РАЗРАБОТКА НАДСТРОЙКИ ДЛЯ MICROSOFT OFFICE

В наше время ни одно предприятие, производящее однородную или специфическую продукцию по той или иной технологии, не обходится без специалиста в области бухгалтерии. Перед бухгалтером, как специалистом, стоит много задач, задач финансового плана. Ему то и дело приходится следить за финансовым состоянием предприятия, за своевременную уплату налогов, выдачу зарплат и другими подобными операциями. Насчитывают десятки, сотни, тысячи, миллионы денежных операций, которые из-за дня в день совершаются на предприятии. И каждый день приходится возиться с большими громоздкими числами, которые, за достоверностью информации, требует расшифровки – прописного варианта.

В данной работе была предпринята попытка создание такого программного продукта для перевода денежных единиц в прописной вариант.

Для осуществление поставленной цели была использована среда Microsoft Visual Studio с применением объектно-ориентированного языка C#. Главное в языке C# – реализация принципов объектно-ориентированного программирования (ООП). Объектно-ориентированная методика неотделима от C#, и все C# – программы в какой-то степени имеют объектную ориентацию.

Разработанная объектно-ориентированная модель позволяет с легкостью совершенствовать разработанное приложение, добавляя новые денежные единицы и языки используя принципы наследования и полиморфизма объектно-ориентированного языка программирования.

Настройка для Microsoft Office Word и Excel, а также объектная модель были реализованы с использование среды Microsoft Visual Studio 2010 и языка C#. Visual Studio представляет собой набор средств разработки на основе компонентов и другие технологии, предназначенные для создания эффективных высокопроизводительных приложений, без малейшего труда. Расширяемость Visual Studio позволяет создавать собственные шаблоны и надстройки.

В результате была разработана объектно-ориентированная модель перевода числа в прописной вариант этого числа и разработана надстройка по внедрение разработанной объектно-ориентированной модели в приложения Microsoft Office Excel и Word. Разработанная надстройка не только позволит облегчить работу бухгалтерских отделов многих предприятий, но также облегчит работу многих людей оперирующих с денежными единицами, требующими прописного представления на том или ином языке, на той или иной валюте.

*А. Р. Качкан*

*Науч. рук. Н. Б. Осипенко,*

*канд. физ.-мат. наук, доцент*

## **РАЗРАБОТКА КОМПОНЕНТОВ СЕРВЕРНОЙ ЧАСТИ ПРИЛОЖЕНИЯ «ПОДБОР РЕШЕНИЙ, СООТВЕТСТВУЮЩИХ ПРЕДНАЗНАЧЕНИЮ ЧЕЛОВЕКА»**

Разрабатываемое клиент-серверное web-приложение является продолжением реализации, описанной в [1]. Разработанная версия клиент-серверного web-приложения обеспечивает предоставление следующих возможностей: пользователю прохождения тестов; разграничения прав пользователя; обработки полученных данных с целью формирования рекомендаций и индивидуального прогноза в рамках таких направлений, как персональный подбор нужных людей, информационных ресурсов и т.п.

В процессе разработки серверной части web-приложения была создана поддержка добавления новых тестов в базу данных и обработка результатов тестов для последующего сохранения и вывода результатов пользователю. Приложение позволяет динамически добавлять тесты на страницу прохождения тестов, добавлять новые тесты в базу данных с целью формирования улучшенных рекомендаций и индивидуального прогноза.

Серверная часть web-приложения создана с помощью PHP фреймворка «Yii 2.0». Данный фреймворк предоставляет следующие возможности: высокую производительность относительно других фреймворков, написанных на PHP, поддержка парадигмы Модель-представление-контроллер, интерфейсов DAO и Active Record для работы с базами данных, кэширование страниц и отдельных фрагментов, ввод и валидация форм, генерация базового PHP-кода для CRUD-операций, поддержка интернационализации, перехват и обработка ошибок, аутентификация и авторизация (RBAC и ACL), возможность подключения сторонних