

## МЕТОДОЛОГИЯ ИСПОЛЬЗОВАНИЯ КОМПОНЕНТОВ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ ПРИ РЕШЕНИИ ПОЛЬЗОВАТЕЛЬСКИХ ЗАДАЧ

**В.В. Шедько**  
*Витебск, ВГУ*

Использование объектно-ориентированного программирования, как при решении пользовательских задач, так и в классическом программировании стало в настоящее время скорее нормой, чем чем-то новым, однако норма эта распространяется на программистов – людей специально подготовленных, профессионалов [3, 4]. В нашем исследовании речь пойдёт об аспектах использования методологии объектно-ориентированного программирования при решении задач обыкновенными пользователями.

Методико-технологическая новизна рассматриваемого вопроса заключается в расширении и модификации использования педагогического дидактического принципа опоры на имеющиеся знания. Данный дидактический принцип успешно используется в содержательных дисциплинах при использовании репродуктивных форм и методов, а для применения данного принципа в технологических дисциплинах требуется его модификация. Модификация данного принципа [1] заключается в его перенесении с имеющихся знаний на имеющиеся умения и навыки.

При распространении данного принципа на использование имеющихся умений и навыков возникает логичный вопрос «Откуда взяться навыкам и умениям в использовании объектно-ориентированного программирования у не подготовленных пользователей?» Действительно, методология объектно-ориентированного программирования в целом эффективна и ориентирована для решения крупных задач и, как следствие, требует специальной, длительной подготовки, что не всегда возможно и целесообразно при решении не больших пользовательских задач, исходя из этого, можно предложить использовать только отдельные компоненты объектно-ориентированного программирования, те из них навыки и умения использования которых имеются у пользователя, а не всю методологию в целом.

Необходимо отметить нюанс, что умения и навыки использования компонентов объектно-ориентированного программирования, как и навыки моделирования, имеются у всех пользователей, просто не все из них догадываются, что выполняя тот или иной мыслительный или технологический приём, они применяют и закрепляют навык или умение применения того или иного компонента объектно-ориентированного программирования. В объяснении пользователю, какой именно компонент, и какой навык объектно-ориентированного программирования он использует для получения решения стоящей задачи и состоит сущность рассматриваемого методического вопроса.

Факт наличия указанных умений и навыков у пользователя объясняется очень просто: окружающий нас мир состоит из объектов, которые обладают определенным набором свойств, у свойств имеются определенные наборы значений, объекты можно классифицировать (относить к определённым классам), над объектами происходят определенные действия, события, существуют определенные связи и т.д., что в свою очередь полностью согласуется с концепцией объектно-ориентированного программирования [2, 4].

Компонентов объектно-ориентированного программирования, на умения и навыки по которым можно опираться при решении пользовательских задач, достаточно много [2]: практически все компоненты объектно-ориентированного

программирования. Все компоненты связаны, с имеющимися у пользователя умениями и навыками: в большей или в меньшей степени, необходимо только правильно реализовать активизацию этих связей.

Рассмотрение методики, форм, методов и различных аспектов применения [1] имеющихся у пользователя умений и навыков по каждому из компонентов объектно-ориентированного программирования при решении задач выходит за рамки данной статьи. В качестве примера можно привести использование навыков и умений относительно компонента «наследование»: при решении любой пользовательской задачи, вначале, составляется каркас (план или «предок») решения, которое потом уточняется: создаются «потомки» решения, которые наследуют свойства и методы родительского решения, приобретая при этом новые свойства и методы, в которых и состоит сущность уточнений.

Перенос дидактического принципа опоры на имеющиеся знания, со знаний на умения и навыки в совокупности с переходом от целостной методологии объектно-ориентированного программирования к отдельному применению её компонентов при решении пользовательских задач позволяет, максимально используя преимущества [2] объектно-ориентированного программирования, существенно сгладить и преодолеть его недостатки [2] в рамках проблемы решения и обучения решению пользовательских задач.

#### ЛИТЕРАТУРА

1. Петров А.Н. Основные подходы к обучению студентов объектно-ориентированному программированию и проектированию // *Фундаментальные исследования*. – 2008. – № 4 – С. 80-82
2. [http:// www.intuit.ru/department/pl/javapl/2/](http://www.intuit.ru/department/pl/javapl/2/).
3. [http://www.znannya.org/Methodology\\_object\\_oriented\\_programming](http://www.znannya.org/Methodology_object_oriented_programming).
4. [http:// www.uchi-it.ru/index.html](http://www.uchi-it.ru/index.html)

### ОПЕРАТОРЫ ДРОБНОГО ИНТЕГРОДИФФЕРЕНЦИРОВАНИЯ ПО АДАМАРУ

**С.А. Шлапаков**

*Витебск, ВГУ*

Дробное интегрирование Римана-Лиувилля является формально дробной степенью  $(d/dx)^\alpha$  оператора дифференцирования  $d/dx$ . Ж. Адамаром была предложена конструкция дробного интегрирования, являющаяся дробной степенью типа  $\left(x \frac{d}{dx}\right)^\alpha$ , так называемого  $\delta$ -дифференцирования

$$\delta = \left(x \frac{d}{dx}\right) [1].$$

На конечном отрезке  $[a, b]$  действительной полуоси рассмотрим интегральный оператор

$$\left(\mathfrak{I}_{a+}^\alpha f\right)(x) = \frac{1}{\Gamma(\alpha)} \int_a^x \frac{f(t)}{\left(\ln \frac{x}{t}\right)^{1-\alpha} t} dt, \quad \alpha > 0, \quad 0 < a < x \quad (1)$$