

**ИСПОЛЬЗОВАНИЕ ВИЗУАЛИЗАЦИИ ПРИ ОБУЧЕНИИ ПРОГРАММИРОВАНИЮ  
THE USE OF VISUALIZATION IN TEACHING PROGRAMMING**

**Дацун Н.Н., Уразаева Л.Ю.**

**Datsun N.N., Urazaeva L.U.**

*Доцент, к.ф.-м.н., Пермский государственный национальный исследовательский университет, Россия, Пермь*

*Доцент, к.ф.-м.н., Сургутский государственный педагогический университет, Россия, Сургут*

*Perm State University, Russia, Perm*

*Surgut State Pedagogical University, Russia, Surgut*

*E-mail: npropovan@gmail.com*

*E-mail: delovoi2004@mail.ru*

**Аннотация:** В статье рассмотрены проблемы обучения программированию, проанализированы особенности освоения данным видом деятельности, предложено использование визуальных инструментов для повышения эффективности обучения программирования и овладения компетенциями программиста.

**Ключевые слова:** особенности обучения программированию, визуальные средства для освоения программирования, уровни мышления при овладении программированием

**Abstract:** The authors describe the problems of teaching programming in the article, they analyze the features of the development of this activity, suggest the use of visual tools to improve the effectiveness of the training program and mastering the competencies programmer

**Keywords:** Problems in the study of programming, visual aids for teaching programming and activation of various levels of thinking when we are learning programming

Программирование является специфическим видом человеческой деятельности, для успешной реализации которой необходимо не только применение приобретенных в процессе обучения знаний и умений, но требуется также наличие определенного стиля мышления, прежде всего, абстрактного, но связанного с необходимостью решения конкретной задачи, умения самостоятельно конструировать сам процесс решения. Программа - это представление алгоритма для решения задачи на языке программирования. Таким образом, решается две задачи: первая задача состоит в составлении алгоритма, вторая задача состоит в переводе алгоритма на понятный для компьютера язык. Причем, как правило, не-

обходимо составить алгоритм для незнакомой для программиста задачи. Также имеются ограничения по времени и среде реализации программы.

Программирование требует проведения анализа предложенной задачи, расчленения ее на простейшие составляющие. Выделение базовых структур алгоритма представляет собой процедуру разложения решения некоторой задачи на последовательность некоторых типовых повторяющихся действий. Затем для построения программы требуется перевод составленного алгоритма на выбранный язык программирования с учетом используемых структур данных и особенностей языка программирования. Сложность програм-

мирования для человека связана с тем, что в повседневной деятельности люди работают с процедурами, представляющимися собой свернутые последовательности шагов, для достижения определенных задач. Большое значение при обучении программированию имеет развитость у человека мышления по аналогии. Мышление по аналогии позволяет нам принимать решения, проводя сравнения с уже известным нам фактом или утверждением. Способность мыслить по аналогии позволяет успешно применять прошлый опыт, выявлять сходство или различия между объектами. Мышление по аналогии позволяют создать или понимать новое, сравнивая его с уже известным. Однако проводя аналогию между процессами или объектами, необходимо верно выбрать основу для сравнений.

Практика показывает, что при обучении программированию учащихся можно разбить на две категории: владеющие методом аналогии и не владеющие методом аналогии. Также учащиеся различаются уровнем развития абстрактного мышления. Самым главным в программировании является способность применения имеющихся формальных знаний и методов мышления для получения готового продукта. Теоретическое мышление формируется у человека к 15-18 годом. До этого преобладают более низкие формы мышления: наглядно-действенное (наблюдается даже у животных) и наглядно-образное.

Так как программирование стало очень широко распространенным, то возникла необходимость приобщения к нему широких слоев населения, в частности школьников. Авторами исследованы различные аспекты обучения программированию, в настоящее время внимание было обращено на визуализацию процесса обучения. Результаты предшествующих исследований изложены в работах [1], [2], [3], [4], [5], [6] и др. Несмотря на многообразие работ, посвященных проблемам обучения программирования с использованием инновационных технологий, развитие ИТ приводит в созданию новых технологий, которые необходимо исследовать с точки зрения их эффективного использования в обучении и целесообразности применения в учебном процессе.

На заре развития программирования наглядные средства разработки алгоритмов применялись очень активно - это были всем известные блок-схемы. С усложнением разрабатываемых информационных систем и программ средства визуализации также развивались и активно использовались.

В настоящее время на рынке имеются несколько известных средств программирова-

ния, опирающихся на возможности активного использования наглядно-образного мышления. Особое место среди них занимают такие средства как Blockly, Drakon.

Важно отметить, что даже визуальный инструмент программирования, всегда основан на использовании абстрактного языка. Некоторые абстрактные понятия легче даются мужчинам, чем женщинам. Это связано с тем, что у мужчин более развито левое полушарие. И им легче представлять теоретически данные и абстрактные объекты для их представления. Однако практика показывает, что при восприятии абстрактных понятий (классы, структуры) молодежь не испытывает проблемы. Гораздо сложнее переучиваться состоявшимся программистам, привыкшим к процедурному программированию. Стереотипы мышления мешают привыкнуть к использованию новых инновационных технологий программирования. Повышение уровня абстрактности легче дается молодежи. Различные формы представления данных даются им достаточно легко. Главным достоинством хорошего программиста можно считать умение создавать универсальные алгоритмы, но при этом решающие успешно конкретные задачи. Как уже отмечалось, программирование сводится к разбиению реальной задачи на элементарные шаги, и затем перевода этих шагов для программной реализации задачи. Разбиение задачи не всегда является легким и успешным. Программист сталкивается с недостатком представления общей картины происходящего. Возможно и усложнение алгоритма при объединении отдельных шагов. Важнейшим моментом является графическое изображение общего плана работы алгоритма. Для представления такой картины необходимо наличие некоторого опыта программирования.

Принято выделять два основных стиля при написании программ: программирование снизу вверх и программирование сверху вниз. Предпочтительнее программирование сверху вниз, в этом случае программист выполняет стратегическое планирование своей деятельности, создает иерархию связанных задач. Основными инструментами визуальной поддержки в работе программиста служат case – технологии.

Незаслуженно мало распространён и используется в России отечественный язык Дракон. Этот язык был разработан для оптимизации труда программистов, автоматизирующих производственные процессы в космической отрасли. Существуют бесплатные редакторы для работы с этим языком. Основной подход к составлению программ в Драконе - это опо-

ра на использование усовершенствованных блок-схем, то есть визуальный (графический) подход к созданию программ.

С точки зрения реализации общего плана решения задачи и последующей реализации отдельных шагов подходит язык Дракон. Программы на языке Дракон можно реализовать с помощью бесплатных редакторов. Отметим, что при использовании Дракона используется абстрактно-логический и наглядно-образный уровни мышления, которые к 15-18 годом не у всех одинаково развиты.

Кроме этого при обучении можно использовать Blockly. Эта среда опирается на использование наглядно – деятельностного уровня мышления, которое в принципе выражено у всех учащихся.

Если у человека имеются проблемы в переводе конструкций алгоритма на язык программирования ему может помочь Blockly. Blockly - это визуальный редактор, который позволяет пользователям писать программы, комбинируя готовые блоки. Можно интегрировать редактор в собственные веб-приложения, чтобы дать возможность начинающим пользователям создать свой интерфейс. Созданную с помощью этого редактора программу перевести на код JavaScript, Python, PHP, Dart, или какой-либо другой язык: Приложение может выполнить код по желанию. Blockly является инструментом для разработчиков приложений. Blockly поставляется с большим количеством предопределенных блоков. Все, начиная от математических функций цикла структуры. Тем не менее, для того, чтобы взаимодействовать с внешним приложением, необходимо создавать пользовательские блоки. В большинстве случаев простой подход, чтобы просто найти действительно похожий блок, который уже существует, скопировать его и изменить его по аналогии.

В сети можно найти достаточное количество примеров и других языков и их редакторов.

Плюсы понятны: при программировании с использованием визуальных объектов человек описывается на образы, обучение активизирует различные уровни мышления. Но минус - может произойти торможение в развитии программирования, в человеческой истории все языки, опирающиеся на иероглифы, приводили к дополнительным сложностям в обучении.

Рекомендации по использованию средств визуальной поддержки программирования:

Использовать средства визуальной поддержки на начальных этапах обучения про-

граммированию (иначе всегда будет чтение «вслух по слогам», а не про себя «чтение массивами»); постоянное использование одной графической поддержки может затормозить развитие новых навыков соединения отдельных задач в единое целое, при отсутствии полной информации;

Стимулировать развитие абстрактно-логического мышления, переходом средства графической поддержки, требующие включения теоретического мышления.

Индивидуальные рекомендации при самостоятельном изучении программирования:

Рассмотреть и оценить несколько различных ресурсов.

Выбрать один оптимальный ресурс и пройти весь курс до конца, чтобы получить базовые знания.

Начать самостоятельные разработки, связанные с практическими задачами (работать в группе, изучать чужой код и литературу, начинать с легких задач)

#### *Библиографический список:*

Дацун Н.Н., Уразаева Л.Ю. Использование массовых открытых онлайн-курсов в математической подготовке специалистов по программной инженерии. Интернет-журнал Науковедение. 2015, № 7(2). с.4.

Галимов И.А., Дацун Н.Н., Уразаева Л.Ю., Уразаева Н.Ю. Интеллектуальная деятельность в сфере разработки ИТ-продуктов для образования. Современные информационные технологии и ИТ-образование. 2014. № 10. С. 261-268.

Дацун Н.Н., Уразаева Л.Ю. Проблемы обучения программированию: взгляд обучаемых непрерывного и базового образования. Электронное обучение в непрерывном образовании. 2014. Т. 1. № 1 (1). С. 55-60.

Уразаева Л.Ю., Дацун Н.Н. Потребности рынка труда и особенности отношения студентов различных направлений подготовки к обучению. Проблемы экономики. 2013. № 3 (55). С. 43-46.

Галимов И.А., Уразаева Л.Ю., Уразаева Н.Ю. Проблемы подготовки к тьюторской деятельности и современные требования на рынке труда. Интернет-журнал Науковедение. 2013. № 3 (16). С. 9.

Уразаева Л.Ю., Галимов И.А. Математическое моделирование e-learning.. Научный вестник Норильского индустриального института. 2009. № 4. С. 15-17.