

О РАЗВИТИИ ВЫЧИСЛИТЕЛЬНОГО МЫШЛЕНИЯ И AGILE-ПРАКТИКАХ В ОБРАЗОВАТЕЛЬНОМ ПРОЦЕССЕ УЧРЕЖДЕНИЙ ВЫСШЕГО ОБРАЗОВАНИЯ

Д. В. ФИЛИМОНОВ¹⁾

¹⁾Белорусский государственный университет, пр. Независимости, 4, 220030, г. Минск, Беларусь

На основе обзора ряда современных методик развития вычислительного мышления описываются особенности интеграции методологии фреймворков *Agile* в образовательный процесс ряда дисциплин естественно-научного цикла учреждений высшего образования.

Ключевые слова: вычислительное мышление; подготовка студентов; естественно-математические специальности; фреймворк; методология *Agile*-практик.

ON THE DEVELOPMENT OF COMPUTATIONAL THINKING AND AGILE-PRACTICES IN THE EDUCATIONAL PROCESS OF UNIVERSITIES

D. V. FILIMONOV^a

^aBelarusian State University, 4 Niezaliežnasci Avenue, Minsk 220030, Belarus

Based on a review of a number of modern methods for the development of computational thinking, the article describes the features of integrating the methodology of *Agile* frameworks into the educational process of a number of disciplines of the natural science cycle of higher educational institutions.

Keywords: computational thinking; preparation of students; natural sciences and mathematics; framework; *Agile*-practices methodology.

Введение

Компьютерные технологии всесторонне интегрируются в современную жизнь, и многие области знания уже имеют прикладные, или вычислительные, ответвления. Идеи С. Пейперта, позже переосмысленные Ж. М. Винг [1], признаны ключевыми в рамках образовательных программ по всему миру, будучи направленными на развитие таких навыков цифрового века, как декомпозиция, обобщение, абстрагирование и дебаггинг.

Под декомпозицией понимается умение разбить сложный объект (проблему) на несколько частей, чтобы процесс нахождения решения стал проще и понятнее. Обобщение подразумевает способность распознавать при анализе задачи общие черты и признаки (шаблоны), которые присущи ряду известных предметов и явлений. Абстрагирование – способность исключить незначительные и несущественные детали, при этом выделить основную

Образец цитирования:

Филимонов ДВ. О развитии вычислительного мышления и *Agile*-практиках в образовательном процессе учреждений высшего образования. *Университетский педагогический журнал*. 2022;2:61–65.

For citation:

Filimonov DV. On the development of computational thinking and *Agile*-practices in the educational process of universities. *University Pedagogical Journal*. 2022;2:61–65. Russian.

Автор:

Дмитрий Васильевич Филимонов – ассистент кафедры веб-технологий и компьютерного моделирования механико-математического факультета.

Author:

Dmitrii V. Filimonov, assistant at the department of web-technologies and computer modelling, faculty of mechanics and mathematics.
dzfilimonau@gmail.com

концептуальную идею. Дебаггинг означает оценку промежуточных результатов.

Ж. М. Винг говорила о сужении подхода в функциях к вычислительному мышлению, так как формирование и развитие данного набора когнитивных навыков применительно лишь к ИТ-специалистам. Первоначальная идея Ж. М. Винг и образовательная практика последних лет свидетельствуют о том, что развитие мыслительных процессов, лежащих в основе вычислительного мышления, позволяет человеку любой профессии эффективнее работать с информацией, а обучение этим навыкам не должно сводиться к подготовке только ИТ-специалистов (что является отдельной проблемой из-за растущей конкуренции) [1–3].

В наибольшей степени такие навыки необходимы людям, которые работают с объемными данными, однако не стоит забывать и о тех, кто отдает предпочтение теоретическим областям знания – в большей мере это касается математиков. Многие идеи, кото-

рые были разработаны несколько столетий назад, получили практическое применение лишь в прошлом веке, особенно с развитием вычислительной техники, позволившей проверять истинность гипотез. Темпы прогресса значительно ускорились, и в условиях четвертой технологической революции недостаток квалификации в работе со специализированными средами становится заметнее. По этой причине особую роль стоит уделять внедрению перспективных направлений и форм обучения с применением информационных технологий, что отвечает концепции и методологии проекта «Цифровой университет», которые сегодня развиваются и реализуются в БГУ. Решение вопроса цифровой грамотности может привести к раскрытию творческого потенциала любого специалиста и появлению открытий (особенно на стыке наук), где ключевую роль играют умения декомпозиции, анализа и критической оценки методов работы с информацией в процессе решения поставленных задач [1–3].

Обзор методик развития вычислительного мышления

Тесная связь вычислительного мышления с набором когнитивных навыков, необходимых математикам, позволяет выделить достаточно широкие классы задач, решение которых требует уверенного владения компьютерными технологиями. Кроме того, как отмечал создатель вычислительной среды *Mathematica* С. Вольфрам, академическое образование является весьма сложным для восприятия теми студентами, которые не имеют достаточного уровня абстрактного мышления [4], а наглядность и когнитивная визуализация, реализуемые средствами программирования, облегчают процесс усвоения.

Многие исследователи уже привели свои результаты внедрения методологии в образовательный процесс вычислительных направлений. Так, М. Тедре и П. Деннинг выделили два блока дисциплин, связанных как с компьютерными технологиями, так и с прикладными математическими задачами, распределив их по объему навыков вычислительного мышления, которые необходимы для успешного освоения соответствующих разделов [5]. Большинство *CT Ideas*¹, рассмотренных в работе [5], входят в число дисциплин, присутствующих в программах математических и инженерных специальностей, и являются отправной точкой для внедрения инноваций в образовательный процесс. Исследование эффективности усвоения материала при сочетании подходов, способствующих развитию вычислительного мышления и STEM-образования применительно к биологии, проведено в работе Х. Свэнсона и его соавторов [6]. Разработанная ими методика может быть использована в других областях знания, особенно это касается первого года университетской

программы как психологически трудного для вчерашних абитуриентов.

Курс численного моделирования (с применением методологии развития вычислительного мышления) в рамках дисциплины «Физика» был разработан рядом ученых [7]. Они изучили результаты, полученные с помощью нескольких подходов: решения задач с вариантами ответов, а также интервью и эссе на изученную тему. Анализ показал, что предложенный курс способствовал синтезу знаний.

В продолжение идей С. Вольфрама группа исследователей [8] изучила то, насколько эффективны методики вычислительного мышления при обучении инженеров на первом курсе. Учет особенностей специальности позволил авторам установить, что задачи обработки данных в таких математических пакетах, как *MATLAB*, действительно помогают систематизировать и улучшить наглядность материала, что является подтверждением идей С. Вольфрама.

Как свидетельствует практика Белорусского государственного университета, после окончания учреждения исключительно теоретическими изысканиями занимается небольшой процент выпускников математических специальностей и многие, даже не имея склонности к программированию, достигают определенных успехов в анализе данных, так как обладают фундаментальными знаниями ряда процессов, стоящих в основе этой области исследований [3].

Несмотря на то что Ж. М. Винг предложила рассматривать идеи С. Пейперта как нечто большее, чем методологию подготовки программистов, молодые специалисты вынуждены адаптироваться к конкурентным условиям с теми специалистами, которые

¹*Computational thinking ideas* – учебные программы и темы, позволяющие продвигать вычислительное мышление.

менее осведомлены в теоретических вопросах, но более эффективны в решении прикладных задач. В таких ситуациях требуются навыки командной работы, способности комплексного применения разных подходов и анализа оптимальности выбранного метода решения, а также навыки анализа, близкие по структурным элементам к вычислительному мышлению [2; 3]. Вместе с тем вычислительное мышление подвергается обоснованной критике ряда крупных специалистов в области математики и информатики, что в своей работе осветил Э. Нарделли [9]. Действительно, внедрение идей Ж. М. Винг может быть осложнено отсутствием соответствующей квалификации у преподавателей и появлением трудностей, которые обусловлены определением места вычислительного мышления в образовании (например, вопрос о том, быть ему отдельной дисциплиной, как это предложено в ряде стран, или вспомогательным инструментом для модернизации уже имеющихся учебных программ).

Особый интерес представляет именно второй путь. Для развития соответствующих современным требованиям навыков понадобятся принципиально новые педагогические фреймворки, дополняющие учебный процесс. Между тем их разработка и апробация могут занять много времени, и часть идей, стоящих за ними, потеряют актуальность. Следовательно, появляется необходимость в применении подходов, прошедших испытание временем и показавших хорошие результаты. Нельзя упускать из виду важность повышения конкурентоспособности молодых специалистов, которые в процессе обучения использовали подобные фреймворки. Методология *Agile* включает все указанные требования.

Возникшие *Agile*-фреймворки (в первую очередь для повышения эффективности коллективов разработчиков в крупных компаниях) основаны на двенадцати принципах², сформулированных ведущими специалистами своей сферы, и обобщены тезисами *Agile*-манифеста:

- люди и их взаимодействие важнее процессов и инструментов;
- работающий продукт важнее исчерпывающей документации;
- сотрудничество с заказчиком важнее согласования условий контракта;
- готовность к изменениям важнее следования первоначальному плану³.

В целях внедрения данных положений в современный образовательный процесс их можно переформулировать (например, как в работе [12]) в качестве следующих основополагающих концепций для проектной формы организации деятельности:

- учет индивидуальных особенностей и уровня развития каждого студента важнее общих образовательных задач;
- достоверность результатов важнее исчерпывающего описания проведенной работы;
- взаимодействие студентов и преподавателей должно быть постоянным на протяжении всего курса (проекта), так как это является обязательным средством контроля и корректировки;
- учет отзывов о проделанной работе важнее следования первоначальному плану.

Фреймворки *Agile*, как следует из названия, гибкие, их основные шаги могут меняться местами в зависимости от возникших проблем и, более того, могут перестать быть шагами как таковыми и выполняться параллельно. По классическому циклу работы согласно *Agile* сначала проводится анализ требований, затем ведется планирование работы, создается и реализуется прототип (эскиз решения), проверяются и корректируются предполагаемые результаты (дебаггинг) и в конце полученное решение предлагается для оценивания.

Таким образом, с дидактической точки зрения *Agile* стимулирует развитие коммуникативных навыков, направленность на командную работу. Деятельность студентов обладает прогностическим характером с поэтапной рефлексией и оценкой результатов. Очевидно, что указанные шаги во многом схожи с теми, которые предлагаются в вычислительном мышлении, при этом широкое распространение и продолжительная практика *Agile*-методологий во всемирно известных компаниях предоставляют большую выборку опыта их внедрения.

Гибкость *Agile*-подходов позволяет организовать работу коллектива соответственно рассматриваемой задаче. Несмотря на то что большинство фреймворков полагаются на итеративное улучшение проекта (с упором на взаимодействие внутри команды, а также команды и преподавателя), можно выделить и несколько подходов, имеющих выраженные особенности.

Во-первых, это динамическая разработка (*Dynamic systems development method*, DSDM). Она подразумевает разделение работы на подготовительный, проектный и оценочный этапы. Частота внесения изменений и регулярный дебаггинг также являются отличительными показателями данной методологии. Такое определение делает DSDM схожей с классической проектной моделью.

Во-вторых, это методология *Kanban*. В ней предлагается создание трех очередей: идей, ожидающих реализации; идей, находящихся в разработке; завершенных идей. Данная методология считается одной из наиболее доступных в освоении, поскольку

²The following 12 principles are based on the agile manifesto // Agile Alliance [Electronic resource]. URL: <https://www.agilealliance.org/agile101/12-principles-behind-the-agile-manifesto/> (date of access: 01.09.2022).

³*Agile*-манифест разработки программного обеспечения [Электронный ресурс]. URL: <https://agilemanifesto.org/iso/ru/manifesto.html> (дата обращения: 05.08.2022).

визуализировать работу можно с помощью доски и пометок. Наиболее близким (хотя и неточным) примером такого подхода можно считать доску с номерами заданий из учебника и место, которое отведено под их решение. Случаи, требующие большой работы над ошибками, также попадают под данную методологию.

В-третьих, методология *Scrum*. Работа над проектом делится на циклы, называемые спринтами. Каждый спринт имеет фиксированные сроки, по истечении которых человек должен предоставить результат. Возникшие в ходе оценки замечания становятся задачами для следующего спринта, процесс трансформации замечаний продолжается до полной готовности. Данная методология считается легкой в освоении и трудной в совершенствовании, потому что на каждом ее этапе нужно собрать коллектив для достижения всеобщей занятости. Представленная организация работы показывает хорошие результаты в случаях, когда долгосрочное планирование затруднено, тем самым возможно построение практических занятий всего курса в рамках *Scrum*-подхода с постепенным обновлением знаний и выделением подзадач.

Идея внедрить фреймворки *Agile* в образовательный процесс естественно-научных специальностей не новая, при этом нужно понимать, что классические *Agile*-фреймворки только отчасти попадают под педагогическое определение данного понятия, следовательно, являются основополагающими принципами для разработки именно педагогических фреймворков.

Заключение

Развитие навыков вычислительного мышления может напрямую зависеть от следования фреймворкам, основанным на *Agile*. В то же время набор когнитивных и социальных навыков у людей при творческой командной работе будет шире, чем у представителей алгоритмического мышления. Кроме того, внедрение методологии *Agile*-практик в образовательный процесс может являться фундаментом трансформации содержания обучения и методики преподавания (с позиции целенаправленного развития вычислительного мышления) с учетом специфики курсов, излагаемых в учреждениях высшего образования, и повышения конкурентоспособности молодых специалистов, знакомых с признанными мировыми практиками в прикладной и исследовательской деятельности.

Вместе с тем наиболее часто в *Agile* критикуется то, что решение задачи важнее сопровождения результата всеми предоставленными элементами (формулы, рассуждения, доказательства, ссылки на работы и теоремы). Это также стоит учитывать при внедрении подобных практик в прикладные и математические дисциплины. Однако никто не обязывает преподавателя во всем следовать принципам *Agile*, так как это всего лишь инструменты для повышения эффективности. Тем не менее данная методология

Исследованию эффективности фреймворков с использованием *Agile*-практик и схожих подходов при обучении математике и естественно-научным дисциплинам посвящены работы [11–15]. Стоит обратить внимание на проект по внедрению методологии *Kanban* [16], а также на работы [17–18], посвященные применению фреймворков, основанных на *Scrum*-подходе.

К примерам заданий и проектов, которые могут быть выполнены в среде, построенной на фреймворках *Agile*, можно отнести некоторые темы дисциплины «Построение и анализ алгоритмов», которая преподается на механико-математическом факультете БГУ. Так, изучение гибридных сортировок и решение задачи коммивояжера иллюстрируют сильные стороны фреймворков в действии. Учет сложности данных, предоставленных для обработки, необходимость предусматривать специальные случаи и возможность отсутствия четкого плана дальнейших действий для студентов как педагогического хода (с постепенным предоставлением новых знаний и доработкой проекта, начатого с изучением темы) могут быть отображены на доске (методология *Kanban*). Таким образом, развитие проекта, начинающееся с заданий первого уровня (готовых алгоритмов), через оценку оптимальности предложенных решений и анализа всего полученного результата с позиции аналитического и компьютерно-вычислительного подхода позволит стимулировать развитие алгоритмического, аналитического и вычислительного мышления с включением элементов творчества.

соответствует ключевым требованиям для подготовки молодых специалистов: она сопряжена с социально-экономическими или образовательными запросами, создает условия для освоения студентами основ научного исследования и соответствует области науки и современному уровню развития компьютерных технологий [2].

На основании разносторонней применимости *Agile*-фреймворков также можно говорить о том, что информационные технологии (будь то среды разработки или вычислительные пакеты) можно охарактеризовать не только как средства упрощения обработки данных, но и как вспомогательные компоненты для развития наиболее важных когнитивных навыков современного специалиста: абстрагирования, декомпозиции, последовательности, анализа и синтеза. Не имеет особого значения то, о специалисте какого направления идет речь, – синтез является неотъемлемой составляющей креативности. При применении *Agile* передача знаний преобразуется в тесное сотрудничество и обмен опытом с учетом индивидуальных особенностей студентов. Обучающиеся вносят свои предложения о том, что стоило бы изменить преподавателю в таком подходе. Преподаватель также оценивает студентов за их работу.

Библиографические ссылки / References

1. Wing J. Computational thinking. *Communications of the ACM*. 2006;49(3):33–35. DOI: 10.1145/1118178.1118215.
2. Brovka NV, Ablameiko SV. [Applied aspects of computer vision in the preparation of undergraduates at a classical university]. In: Noskov MV, editor. *Informatizatsiya obrazovaniya i metodika elektronnoy obucheniya: tsifrovye tekhnologii v obrazovanii. Materialy V Mezhdunarodnoi nauchnoi konferentsii; 21–24 sentyabrya 2021 g.; Krasnoyarsk, Rossiya* [Informatisation of education and e-learning methodology: digital technologies in education. Materials of the 5th International Scientific Conference; 2021 September 21–24; Krasnoyarsk, Russia]. Krasnoyarsk: Siberian Federal University; 2021. p. 78–82. Russian.
3. Ablameyko MS, Brovka NV. Should students of all majors study artificial intelligence? *Annals of Social Sciences and Management Studies*. 2022;7(3):555711. DOI: 10.19080/ASM.2022.07.555711
4. Wolfram S. *How to teach computational thinking* [Internet]. 2016 [cited 2022 July 23]. Available from: <https://writings.stephenwolfram.com/2016/09/how-to-teach-computational-thinking>.
5. Denning PJ, Tedre M. Computational thinking: a disciplinary perspective. *Informatics in Education*. 2021;20(3):361–390. DOI: 10.15388/infedu.2021.21.
6. Swanson H, Irgens GA, Bain C, Hall K, Woods P, Rogge C, et al. Characterizing computational thinking in high school science. *International Society of the Learning Sciences*. 2018;3:871–878. DOI: 10.22318/cscl2018.871.
7. Aiken JM, Caballero MD, Douglas SS, Burk BJ, Scanlon E, Thoms BD, et al. Understanding student computational thinking with computational modeling. In: Engelhardt PV, Churukian A, Rebello NS, editors. *Physics education research conference. AIP conference proceedings; 2012 August 1–2; Philadelphia, USA* [Internet]. 2012 [cited 2022 September 1]. Available from: <https://aip.scitation.org/doi/pdf/10.1063/1.4789648>.
8. Cruz Castro LM, Magana A, Douglas K, Boutin M. Analyzing students' computational thinking practices in a first-year engineering course. *IEEE Access*. 2021;9:33041–33050. DOI: 10.1109/ACCESS.2021.3061277.
9. Nardelli E. Do we really need computational thinking? *Communications of the ACM*. 2019;62(2):32–35. DOI: 10.1145/3231587.
10. Stewart JC, DeCusatis CS, Kidder K, Massi JR, Kirk MA. *Evaluating agile principles in active and cooperative learning* [Internet]. 2009 [cited 2022 August 1]. Available from: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.510.3904&rep=rep1&type=pdf>. DOI: 10.1.1.510.3904.
11. Fronza I, El IN, Corral L. Teaching computational thinking using agile software engineering methods: a framework for middle schools. *ACM Transactions on Computing Education*. 2017;17(4):1–28. DOI: 10.1145/3055258.
12. Moreira CO. The development of computational thinking in computing higher education. In: Association for Computing ACM Technical Machinery. *GIGCSE 2022: the 53rd ACM technical symposium on computer science education; 2022 March 2–5; Providence, USA*. New York: Association for Computing Machinery; 2022. p. 1172. DOI: 10.1145/3478432.3499217.
13. Dewi DA, Muniandy M. The agility of agile methodology for teaching and learning activities. In: Dayang Norhayati Abang Jawawi, Shahida Sulaiman, Nor Azizah Sa'adon, Radziah Mohamad, editors. *2014 8th Malaysian software engineering conference; 2014 September 23–24; Langkawi, Malaysia*. Malaysia: Institute of Electrical and Electronics Engineers; 2014. p. 255–259. DOI: 10.1109/MySec.2014.6986024.
14. Duvall S, Hutchings D, Kleckner M. Changing perceptions of discrete mathematics through scrum-based course management practices. *Journal of Computing Sciences in College*. 2017;33(2):182–189. DOI: 10.5555/3144645.3144672.
15. Noguera I, Guerrero A-E, Maso R. Collaborative agile learning in online environments: Strategies for improving team regulation and project management. *Computers & Education*. 2017;116:110–129. DOI: 10.1016/j.compedu.2017.09.008.
16. Bacea I, Ciupe A, Meza S. Interactive Kanban – blending digital and physical resources for collaborative project based learning. In: Chang Maiga, Chen Nian-Shing, Huang Ronghuai, Kinshuk, Sampson Demetrios G, Vasiliu Radu, editors. *IEEE 17th International conference on advanced learning technologies; 2017 July 3–7; Timisoara, Romania*. Timisoara: Institute of Electrical and Electronics Engineers; 2017. p. 210–211. DOI: 10.1109/ICALT.2017.68.
17. Delhij A, Solingen R, Wijnands W. *The eduScrum guide* [Internet]. 2015 [cited 2022 September 1]. Available from: http://agileineducation.ru/wp-content/uploads/2018/01/The_eduScrum_Guide_EN_1.2.pdf.
18. Gestwicki P, McNely B. Interdisciplinary projects in the academic studio. *ACM Transactions on Computing Education*. 2016;16(2):1–24. DOI: 10.1145/2732157.

Статья поступила в редколлегию 07.09.2022.

Received by editorial board 07.09.2022.