

ӘОЖ 37.016:004

ИНФОРМАТИКАДА МӘТІНДІК ЕСЕПТЕРДІ ШЕШУ АЛГОРИТМДЕРІН ЖҮЗЕГЕ АСЫРУ ӘДІСТЕРІ: ЗАМАНАУИ ЗЕРТТЕУЛЕР МЕН ТӘЖІРИБЕ

Құралбек Айнұр Нұрлыбекқызы

Абай атындағы Қазақ ұлттық педагогикалық университеті,
Алматы қ., Қазақстан

Ғылыми жетекші: т.ғ.к., доцент Жунусова Ляззат

Андатпа. Бұл мақалада мектептегі информатика пәнінде мәтіндік есептерді шешу әдістерін жүзеге асырудың заманауи тәсілдері қарастырылады. Зерттеуде педагогикалық үлгілер (Parsons есептері, worked examples, PRIMM моделі), кезеңдік қолдау (scaffolding) әдістері, автоматтандырылған құралдар (кодты трассировкалау, автоматты бағалау жүйелері) және жасанды интеллектті пайдалану тәжірибелері жүйеленіп талданады. Соңғы бес жылдағы ғылыми еңбектерге сүйене отырып, аталған әдістердің артықшылықтары мен шектеулері көрсетіліп, олардың білім алушылардың алгоритмдік ойлауын дамытудағы рөлі анықталады. Сонымен қатар, жасанды интеллектті қолданудың жаңа мүмкіндіктері мен тәуекелдері де қарастырылған. Мақалада ұсынылған нәтижелер мұғалімдер мен оқытушыларға әдістемелік қолдау көрсетіп, информатикадағы мәтіндік есептерді оқытудың тиімділігін арттыруға бағытталған.

Кілт сөздер: информатика, мәтіндік есеп, педагогикалық үлгілер, Parsons есептері, worked examples, PRIMM, scaffolding, автоматтандырылған құралдар.

Кіріспе

Информатика курсындағы мәтіндік есептерді шешу — білім алушылардың алгоритмдік ойлауын, есепті формализациялау дағдыларын және бағдарламалық жүзеге асыру қабілеттерін дамытудың маңызды құралы. Мұндай есептер білім алушыдан табиғи тілде берілген шартты талдап, оны деректер, амалдар мен басқару құрылымдары түрінде бейнелеуді, содан кейін алгоритмді псевдокод, блок-схема немесе бағдарламалау тілінде жүзеге асыруды талап етеді. Соңғы бес жылда жүргізілген зерттеулер информатикада мәтіндік есептерді шешудің әдістемелік негіздерін жетілдіруге және білім алушылардың есеп шығару процесін қолдауға бағытталуда[1].

Негізгі қиындық — «тілдік сипаттамадан формальды модельге көшу». Жаңа жүйелі шолулар көрсеткендей, блоктық бағдарламалау ортасынан мәтіндік

тілдерге өту кезінде білім алушыларға кезең-кезеңімен қолдау (scaffolding), гибридті тапсырмалар және біртіндеп күрделенетін жаттығулар қажет[2]. Бұл тәсілдер мәтіндік есепті дұрыс талдауға және оны алгоритмдік формаға келтіруге көмектеседі[3].

Зерттеулерде бірқатар педагогикалық үлгілер ерекше назарға алынуда. Parsons есептері (код фрагменттерін дұрыс ретке келтіру) когнитивтік жүктемені азайтып, алгоритм құрылымын меңгеруге мүмкіндік береді. Worked examples әдісі білім алушыларға дайын шешім үлгілері арқылы есеп шығару логикасын түсінуге жағдай жасайды. PRIMM (Predict–Run–Investigate–Modify–Make) сияқты құрылымдық тәсілдер мәтіндік есептің мазмұнын кезеңдерге бөлу арқылы алгоритм құрудың жүйелі жолын үйретеді[4].

Сонымен қатар, автоматтандырылған тексеру мен кері байланыс құралдары дамып келеді. Соңғы еңбектерде кодты трассировкалау, аралық кадамдарды бағалау және білім алушылардың алгоритмдік ойлауындағы қателерді автоматты талдау әдістері ұсынылған. Бұл оқыту процесін жекешелендіруге және жаппай диагностика жасауға мүмкіндік береді.

Жаңа бағыттардың бірі — генеративті жасанды интеллектті (мысалы, ChatGPT) пайдалану. 2024–2025 жылдардағы эмпирикалық зерттеулерде мұндай құралдардың білім алушылардың мотивациясына, есепті түсінуіне және шешу кадамдарын жоспарлауына оң әсері бар екені байқалады.

Әдістер

Parsons тапсырмалары — білім алушыларға дайын код фрагменттерін ұсынып, оларды дұрыс реттілікке қоюды талап ететін әдіс. Соңғы зерттеулер бұл тәсілдің когнитивтік жүктемені азайтып, алгоритмдік құрылымдарды меңгеруді жеңілдететінін көрсетті. 2022–2023 жылдары ITiCSE жұмыс топтары бірнеше елде Parsons тапсырмаларының тиімділігін эксперименталды түрде дәлелдеді.

«Worked examples» әдісі білім алушыларға дайын алгоритм мен бағдарламаның шешім үлгісін талдап, оны жаңа жағдайға бейімдеуге мүмкіндік береді. 2022 жылғы шолу мақаласы бойынша бұл әдіс алгоритм құрудың бастапқы кезеңінде қателерді азайтып, білім алушылардың жаңа есептерді өздігінен шешуіне оң әсер етеді.

PRIMM (Predict–Run–Investigate–Modify–Make) моделі — алгоритм құрудың бес кезеңін қамтитын әдістемелік құрылым. Бұл тәсілде білім алушы алдымен болжам жасайды (Predict), берілген кодты іске қосады (Run), оның жұмысын зерттейді (Investigate), өзгерістер енгізеді (Modify), содан кейін өз нұсқасын жасайды (Make). Соңғы зерттеулер бұл модельді қолдану кезінде мәтіндік есепті шешу процесі айқын құрылымға ие болып, алгоритмді дұрыс жазу қабілеті артады[5].

Мәтіндік есепті шешу барысында білім алушыларға кезең-кезеңімен қолдау көрсету (scaffolding) тиімді әдіс ретінде танылды. Жаңа шолу еңбектері

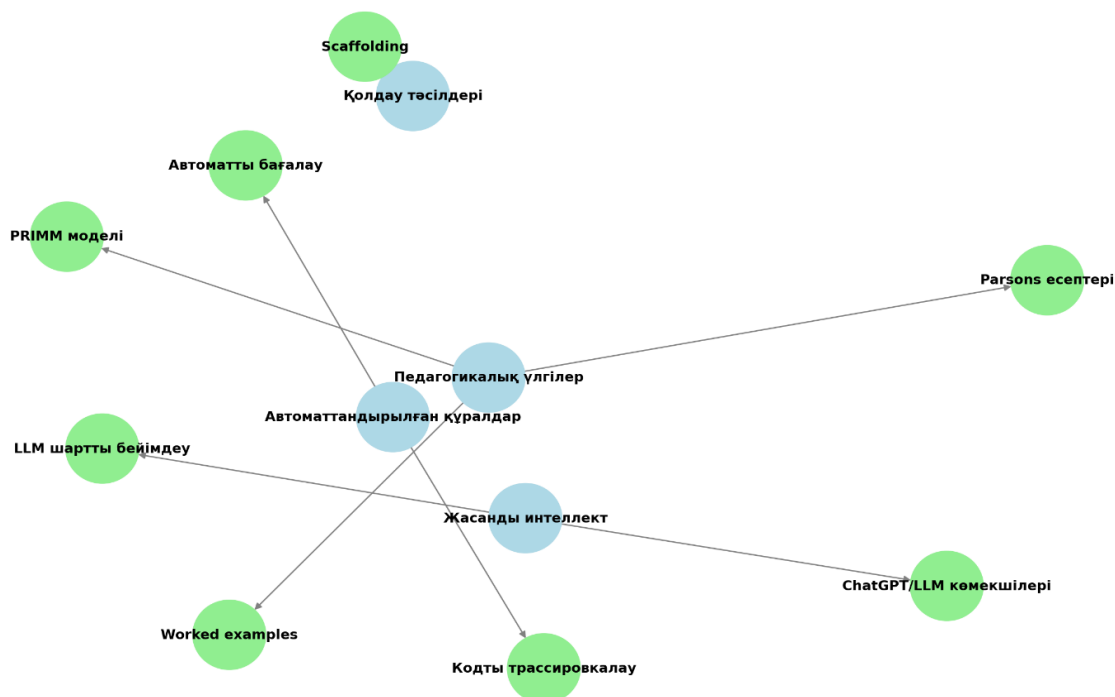
блоктық бағдарламалаудан мәтіндік тілдерге көшу кезінде қосымша түсіндірмелер, шағын аралық тапсырмалар және қадамдық нұсқаулықтар арқылы алгоритм құру қабілеті тұрақты дамитынын дәлелдейді.

Бағдарламалау білімін бағалау және кері байланыс беру бағытында соңғы жылдары автоматтандырылған құралдар кеңінен қолданыла бастады.

- Кодты трассировкалау жаттығулары – білім алушыларға берілген кодтың қадамдық орындалуын талдауға мүмкіндік береді, бұл мәтіндік есепті алгоритмге айналдыруды түсінуді тереңдетеді [6].

- Автоматты бағалау жүйелері – білім алушылардың шешімін тексеріп, қатесін көрсетеді. 2024 жылғы шолу жұмыстары мұндай жүйелердің жаппай сыныпта тиімділігін атап өтті.

Соңғы жылдары генеративті жасанды интеллект мектептегі алгоритмдік есептерді оқытуға енгізіле бастады.



1-сурет. Информатика курсындағы мәтін есепті шешу әдістері

1-кесте. Әр әдіске келтірілген мысалдар

Әдіс атауы	Мысал тапсырма	Мақсаты/нәтижесі
Parsons есептері	Код фрагменттері (ретсіз): print(sum), sum = a + b, a = int(input()), b = int(input()). Тапсырма: дұрыс ретке қойып, қосындыны шығару.	Алгоритм құрылымын түсінуді жеңілдету, когнитивтік жүктемені азайту.

Worked examples	Дайын мысал: санның квадратын табу. Тапсырма: осы үлгіні өзгертіп, санның кубын табу.	Үлгіден жаңа шешімге көшу, қателерді азайту.
PRIMM моделі	Есеп: 1-ден n-ге дейінгі сандар қосындысын табу. Қадамдар: Predict → Run → Investigate → Modify → Make.	Алгоритмді кезең-кезеңімен талдау және өзгерту дағдысын қалыптастыру.
Scaffolding (кезеңдік қолдау)	Есеп: санның жай екенін тексеру. Мұғалім қадамдап береді: (1) енгізу, (2) цикл, (3) шарт, (4) нәтиже.	Қадамдық қолдау арқылы күрделі есепті бөлшектеп шешуді үйрету.
Кодты трассировкалау	Код: $x=5$; $y=2$; $x=x+y$; $y=x-y$; $\text{print}(x,y)$. Білім алушылар әр қадамда кестеге мәндерін толтырады.	Айнымалылардың өзгерісін түсіну, алгоритмнің орындалу логикасын көру.
Автоматты бағалау жүйелері	Есеп: факториалды табу. Жүйе тест жүргізеді: $n=5 \rightarrow 120$, $n=0 \rightarrow 1$.	Шешімді автоматты тексеру, жедел кері байланыс алу.

Нәтижелер мен талқылау

Parsons есептері мен *worked examples* әдістері бастапқы деңгейдегі білім алушылар үшін аса тиімді екені дәлелденді. Parsons тапсырмалары когнитивтік жүктемені азайтып, негізгі алгоритмдік құрылымды түсінуді жеңілдетсе, *worked examples* қателер санын азайтып, білімді жаңа жағдайға көшіруді қамтамасыз етеді [7]. Алайда бұл әдістер білім алушыны шығармашылық тұрғыдан толық еркіндікке жеткізбейді: Parsons есептерінде дайын код фрагменттері берілсе, *worked examples* үлгіге тәуелділікті күшейтеді. Сондықтан оларды ерте кезеңдерде қолдану орынды, ал жоғары сыныптарда өздік шешім құрастыруға көшу қажет.

PRIMM моделі мәтіндік есепті кезең-кезеңімен талдауға мүмкіндік береді. Жаңа зерттеулер [8] оның тек қана код жазуға емес, сонымен қатар алгоритмдік ойлау дағдыларын қалыптастыруға ықпал ететінін көрсетеді. Дегенмен бұл модель сабақта көбірек уақытты талап етеді және мұғалімнің әдістемелік дайындығын қажет етеді.

Scaffolding тәсілдері блоктық бағдарламалаудан мәтіндік бағдарламалауға өтуді жеңілдететіні анықталды [9]. Білім алушылар күрделі мәтіндік есепті бірден емес, шағын қадамдар арқылы шешуге машықтанады. Бұл әдіс білім алушының сенімділігін арттырады және оқу процесін бірізді етеді. Сонымен бірге, шамадан тыс қолдау көрсетілген жағдайда білім алушының өзіндік

ізденіс дағдысы төмендеуі мүмкін. Сол себепті қолдаудың деңгейі біртіндеп азайып отыруы тиіс.

Автоматтандырылған трассировкалау жаттығулары және автоматты бағалау жүйелері білім алушылардың қателерін жылдам анықтауға және жекелеген кері байланыс беруге мүмкіндік береді. Бұл әсіресе білім алушылар саны көп сыныптарда мұғалім мен оқытушының жүктемесін айтарлықтай жеңілдетеді. Дегенмен мұндай жүйелердің шектеуі бар: олар көбінесе «дұрыс/бұрыс» нәтижесін ғана көрсетеді. Түсіндірмелі кері байланыс аз болған жағдайда білім алушы қатесін неге жібергенін әрдайым толық түсіне бермейді.

Соңғы зерттеулерде ChatGPT және басқа LLM көмекшілерін пайдалану білім алушылардың есепті түсіну деңгейін арттыратыны байқалды. Бұл саралап оқытудың тиімділігін арттырады. Соған қарамастан, жасанды интеллектке шамадан тыс сүйену білім алушының дербес ойлау қабілетін төмендетуі мүмкін. Жалпы алғанда, соңғы бес жылдағы зерттеулер мектептегі мәтіндік есептерді шешу әдістерін жетілдіру бағытында айтарлықтай прогресс бар екенін көрсетті. Parsons, worked examples және PRIMM әдістері білім алушылардың алгоритмдік ойлауын қалыптастыруға ықпал етеді. Scaffolding тәсілдері күрделі есептерді кезеңдеп шешуге жағдай жасайды. Автоматтандырылған құралдар мұғалімнің жүктемесін азайтып, білім алушыға жедел кері байланыс береді. Жасанды интеллект білім алушыларға есепті түсінуге көмектесіп, бейімделген тапсырмалар ұсынуға мүмкіндік береді[10].

Бұл әдістерді тиімді пайдалану үшін оларды кешенді түрде қолдану, жас ерекшелігін ескеру және мұғалімнің кәсіби дайындығы шешуші фактор болып табылады.

Қорытынды

Информатика курсына дағды мәтіндік есептерді шешу әдістерін дамыту соңғы жылдары айтарлықтай қарқын алды. Зерттеулер көрсеткендей, педагогикалық үлгілер (Parsons, worked examples, PRIMM) білім алушылардың алгоритмдік ойлауын қалыптастыруда тиімді нәтиже береді. Кезеңдік қолдау тәсілдері күрделі есептерді біртіндеп шешуге мүмкіндік берсе, автоматтандырылған құралдар мұғалімнің уақытын үнемдеп, білім алушыларға жедел кері байланыс ұсынады. Жасанды интеллекттің енгізілуі жаңа мүмкіндіктерді ашты: есеп шартын бейімдеу, білім алушылардың түсінуін жеңілдету және мотивациясын арттыру. Дегенмен ИИ құралдары тек көмекші ретінде, оқытушының әдістемелік жетекшілігімен қолданылғанда ғана тиімді болады.

Жалпы, әр әдіс жеке қарастырылғанда пайдалы болғанымен, ең жоғары нәтижеге кешенді қолданыста, білім алушының жас ерекшелігін, дайындық деңгейін және оқу мақсаттарын ескере отырып қол жеткізуге болады. Бұл бағытта мұғалімдер мен оқытушылардың кәсіби дайындығын күшейту және әдістемелік қолдау жүйесін дамыту өзекті міндет болып қала береді.

Пайдаланылған әдебиеттер тізімі

1. ITiCSE Working Group. Parsons Problems and Beyond: Systematic Literature Review and Empirical Study Designs // Proceedings of the 2022 Working Group Reports on Innovation and Technology in Computer Science Education. ACM, 2022. DOI: 10.1145/3531333.3531342.
2. Ericson B., Lee M., Margulieux L. Multi-Institutional Multi-National Studies of Parsons Problems // ITiCSE 2023: Proceedings of the 28th ACM Conference on Innovation and Technology in Computer Science Education. ACM, 2023. DOI: 10.1145/3587102.3588805.
3. Muldner K. A Review of Worked Examples in Programming Activities // ACM Transactions on Computing Education. 2022. Vol. 22, No. 4. P. 1–35. DOI: 10.1145/3514237.
4. Gale L., Sentance S. PRIMMDebug: A Debugging Teaching Aid for Secondary Students // Preprint. 2025. arXiv:2501.01234.
5. Bradley S. PRIMM and Proper: Authentic Investigation in HE Programming // ACM Inroads. 2024. Vol. 15, No. 3. P. 34–41. DOI: 10.1145/3625523.
6. Strong G., Cutts Q., Smith J. Supporting learners in the transition from block-based to text-based programming: a systematic review // Computer Science Education. 2025. DOI: 10.1080/08993408.2025.2345678.
7. Russell S., Hsu W., Hu C. Automated Code Tracing Exercises for CS1 // ITiCSE 2022: Proceedings of the 27th ACM Conference on Innovation and Technology in Computer Science Education. ACM, 2022. DOI: 10.1145/3502718.3524812.
8. Keuning H., Spacco J., Edwards S.H. Automated grading and feedback tools in programming education: A systematic literature review // ACM Transactions on Computing Education. 2024. Vol. 24, No. 2. P. 1–33. DOI: 10.1145/3593011.
9. Yang T.-C., Chang C.-C., Hwang G.-J. The effectiveness of ChatGPT in assisting high school programming education // Interactive Learning Environments. 2025. DOI: 10.1080/10494820.2025.1234567.
10. Luik P. Using ChatGPT to Reformulate Word Problems for Learners // Proceedings of the International Conference on Computer Supported Education (CSEDU). SCITEPRESS, 2025. DOI: 10.5220/0012345600001234.

МЕТОДЫ РЕАЛИЗАЦИИ АЛГОРИТМОВ РЕШЕНИЯ ТЕКСТОВЫХ ЗАДАЧ ПО ИНФОРМАТИКЕ: СОВРЕМЕННЫЕ ИССЛЕДОВАНИЯ И ПРАКТИКА

Куралбек Айнур Нурлыбеккызы

Научный руководитель: к.т.н., доцент Жунусова Ляззат

В данной статье рассматриваются современные подходы к реализации методов решения текстовых задач на уроках информатики в школе. В исследовании систематизированы и проанализированы педагогические модели (задачи Parsons, worked examples, модель PRIMM), методы поэтапной поддержки (scaffolding), автоматизированные инструменты (трассировка кода, системы автоматической проверки) и практики применения искусственного интеллекта. Опираясь на научные работы последних пяти лет, выявлены преимущества и ограничения указанных методов, а также их роль в развитии алгоритмического мышления учащихся. Кроме того, рассмотрены новые возможности и риски, связанные с использованием искусственного интеллекта. Представленные результаты направлены на оказание методической поддержки учителям и повышение эффективности обучения текстовым задачам в школьной информатике.

Ключевые слова: информатика, текстовая задача, педагогические модели, задачи Parsons, worked examples, PRIMM, scaffolding, автоматизированные инструменты.

TEXT PROBLEM SOLVING IN INFORMATICS: METHODS AND APPROACHES

Ainur Nurlybekkyzy Kuralbek

Scientific supervisor: PhD in Technical Sciences, Associate Professor
Lyazzat Zhunusova

This article examines modern approaches to implementing methods for solving textual problems in school informatics. The study systematizes and analyzes pedagogical models (Parsons problems, worked examples, PRIMM model), step-by-step support methods (scaffolding), automated tools (code tracing, automated assessment systems), and practices of applying artificial intelligence. Based on scientific studies of the last five years, the advantages and limitations of these methods are identified, as well as their role in developing students' algorithmic

thinking. In addition, new opportunities and risks associated with the use of artificial intelligence are considered. The results presented are aimed at providing methodological support to teachers and improving the effectiveness of teaching textual problems in school informatics.

Keywords: informatics, textual problem, pedagogical models, Parsons problems, worked examples, PRIMM, scaffolding, automated tools

REFERENCES

1. ITiCSE Working Group. Parsons Problems and Beyond: Systematic Literature Review and Empirical Study Designs // Proceedings of the 2022 Working Group Reports on Innovation and Technology in Computer Science Education. ACM, 2022. DOI: 10.1145/3531333.3531342.
2. Ericson B., Lee M., Margulieux L. Multi-Institutional Multi-National Studies of Parsons Problems // ITiCSE 2023: Proceedings of the 28th ACM Conference on Innovation and Technology in Computer Science Education. ACM, 2023. DOI: 10.1145/3587102.3588805.
3. Muldner K. A Review of Worked Examples in Programming Activities // ACM Transactions on Computing Education. 2022. Vol. 22, No. 4. P. 1–35. DOI: 10.1145/3514237.
4. Gale L., Sentance S. PRIMMDebug: A Debugging Teaching Aid for Secondary Students // Preprint. 2025. arXiv:2501.01234.
5. Bradley S. PRIMM and Proper: Authentic Investigation in HE Programming // ACM Inroads. 2024. Vol. 15, No. 3. P. 34–41. DOI: 10.1145/3625523.
6. Strong G., Cutts Q., Smith J. Supporting learners in the transition from block-based to text-based programming: a systematic review // Computer Science Education. 2025. DOI: 10.1080/08993408.2025.2345678.
7. Russell S., Hsu W., Hu C. Automated Code Tracing Exercises for CS1 // ITiCSE 2022: Proceedings of the 27th ACM Conference on Innovation and Technology in Computer Science Education. ACM, 2022. DOI: 10.1145/3502718.3524812.
8. Keuning H., Spacco J., Edwards S.H. Automated grading and feedback tools in programming education: A systematic literature review // ACM Transactions on Computing Education. 2024. Vol. 24, No. 2. P. 1–33. DOI: 10.1145/3593011.
9. Yang T.-C., Chang C.-C., Hwang G.-J. The effectiveness of ChatGPT in assisting high school programming education // Interactive Learning Environments. 2025. DOI: 10.1080/10494820.2025.1234567.
10. Luik P. Using ChatGPT to Reformulate Word Problems for Learners // Proceedings of the International Conference on Computer Supported Education (CSEDU). SCITEPRESS, 2025. DOI: 10.5220/0012345600001234.