SCRATCH DASTURLASH MUHITI YORDAMIDA 5–6-SINF O‘QUVCHILARIDA ALGORITMIK TAFAKKURNI RIVOJLANTIRISH METODIKASI

O.T.Murodov[1]

1. “Osiyo xalqaro universiteti” “Aniq fanlar va raqamli texnologiyalar” kafedrasi assistent o’qituvchisi

Mas'ul muallif: Murodov Oybek To’raqulovich, “Osiyo xalqaro universiteti” “Aniq fanlar va raqamli texnologiyalar” kafedrasi assistent o’qituvchisi.

Received: 05 Noyabr 2026
Accepted: 10 Noyabr 2026
Published: 15 Noyabr 2026

Copyright: © 2026 by the author(s). Licensee Buxoro xalqaro universiteti ilmiy axborotnomasi.

Iqtibos uchun: Murodov, O. T. SCRATCH DASTURLASH MUHITI YORDAMIDA 5–6-SINF O‘QUVCHILARIDA ALGORITMIK TAFAKKURNI RIVOJLANTIRISH METODIKASI / O. T. Murodov // Buxoro xalqaro universiteti ilmiy axborotnomasi. — 2026. — T. 11, № 1. — B. 494–499.

Annotatsiya. Maqolada umumiy o‘rta ta’lim maktablarining 5–6-sinflarida “Informatika va axborot texnologiyalari” fanini o‘qitishda Scratch vizual dasturlash muhitidan foydalanib, o‘quvchilarda algoritmik tafakkurni rivojlantirish metodikasi asoslab berilgan. Algoritmik tafakkurning tarkibiy qismlari (dekompozitsiya, qonuniyatlarni aniqlash, abstraksiyalash, algoritm tuzish va nosozliklarni tuzatish) 5–6-sinf o‘quvchilarining yosh xususiyatlari bilan bog‘liq holda tahlil qilingan. “Foydalan – o‘zgartir – yarat – anglab yet” tamoyiliga asoslangan to‘rt bosqichli metodik model, o‘quv modulining mazmuni, loyihaviy topshiriqlar va dars namunasi hamda o‘quvchilarning algoritmik tafakkur darajasini baholash rubrikasi taklif etilgan. Pedagogik tajriba-sinovni tashkil etish va uning natijalarini statistik qayta ishlash tartibi bayon qilingan. Maqola materiallaridan informatika fani o‘qituvchilari, metodistlar va pedagogika yo‘nalishidagi tadqiqotchilar foydalanishi mumkin.

Kalit so'zlar: algoritmik tafakkur, Scratch, vizual dasturlash, informatika, 5–6-sinflar, loyihaviy o‘qitish, innovatsion yondashuv, baholash rubrikasi.

МЕТОДИКА РАЗВИТИЯ АЛГОРИТМИЧЕСКОГО МЫШЛЕНИЯ У УЧАЩИХСЯ 5–6 КЛАССОВ С ПОМОЩЬЮ СРЕДЫ ПРОГРАММИРОВАНИЯ SCRATCH

Аннотация. В статье обосновывается методика развития алгоритмического мышления у учащихся 5–6 классов общеобразовательной школы при обучении предмету «Информатика и информационные технологии» с использованием среды визуального программирования Scratch. Компоненты алгоритмического мышления (декомпозиция, выявление закономерностей, абстрагирование, разработка алгоритма, отладка) проанализированы с учётом возрастных особенностей учащихся. Предложены четырёхэтапная методическая модель на основе принципа «используй – измени – создай – осмысли», содержание учебного модуля, примеры проектных заданий, образец урока и рубрика оценивания уровня алгоритмического мышления. Описан порядок организации педагогического эксперимента и статистической обработки его результатов. Материалы статьи могут быть полезны учителям информатики, методистам и исследователям в области педагогики.

Ключевые слова: алгоритмическое мышление, Scratch, визуальное программирование, информатика, 5–6 классы, проектное обучение, инновационный подход, рубрика оценивания.

METHODOLOGY FOR DEVELOPING ALGORITHMIC THINKING IN GRADES 5–6 STUDENTS USING THE SCRATCH PROGRAMMING ENVIRONMENT

Abstract. The article substantiates a methodology for developing algorithmic thinking in grade 5–6 students in the course “Informatics and Information Technologies” using the Scratch visual programming environment. The components of algorithmic thinking (decomposition, pattern recognition, abstraction, algorithm design and debugging) are analysed in relation to the age characteristics of students. The paper proposes a four-stage methodological model based on the “use – modify – create – reflect” principle, the content of a teaching module, project tasks, a sample lesson and a rubric for assessing the level of algorithmic thinking. The procedure for organising a pedagogical experiment and for the statistical processing of its results is described. The materials may be useful for informatics teachers, methodologists and education researchers.

Keywords: algorithmic thinking, Scratch, visual programming, informatics, grades 5–6, project-based learning, innovative approach, assessment rubric.

Kirish. Zamonaviy jamiyatning raqamli transformatsiyasi umumta’lim maktablari oldiga o‘quvchilarni nafaqat tayyor dasturiy vositalardan foydalanishga, balki ularning ishlash mantig‘ini tushunishga va masalalarni mustaqil yechishga o‘rgatish vazifasini qo‘ymoqda. Ta’lim jarayoniga zamonaviy axborot va pedagogik texnologiyalarni joriy etish, yoshlarning raqamli ko‘nikmalarini rivojlantirish davlat siyosatida ustuvor vazifa sifatida e’tirof etilgan [1; 2].

Bu jarayonda “hisoblash tafakkuri” (computational thinking) tushunchasi alohida

o‘rin tutadi. J. Wing uni o‘qish, yozish va hisoblash qatorida har bir bolaning tahliliy qobiliyatlariga qo‘shilishi lozim bo‘lgan asosiy ko‘nikma sifatida ta’riflagan [3]. Bizning tadqiqotimizda mazmunan yaqin, ammo maktab informatikasi uchun qulayroq bo‘lgan “algoritmik tafakkur” atamasi asosiy tushuncha sifatida qo‘llanadi.

Amaliyotda 5–6-sinf informatika darslarining katta qismi ofis dasturlari va internet bilan ishlash ko‘nikmalarini shakllantirishga qaratiladi, algoritmlash va dasturlash elementlari esa ko‘pincha sxematik tarzda, kompyuterdan ajralgan holda o‘rgatiladi. Matnli dasturlash tillarida ishlash 10–12 yoshli o‘quvchilar uchun sintaksis xatolari, ingliz tilidagi kalit so‘zlar va mavhum yozuv tufayli qiyinlashadi, natijada qiziqish susayadi. Vizual blokli muhitlar bu to‘siqni olib tashlaydi: o‘quvchi diqqatini “qanday yozish kerak” degan savoldan “nima va qanday tartibda bajarish kerak” degan savolga qaratadi.

Scratch – Massachusets texnologiya institutining (MIT) Media Lab laboratoriyasida ishlab chiqilgan va 2007-yildan beri keng qo‘llanilayotgan bepul vizual dasturlash muhiti bo‘lib, S. Papertning konstruksionizm g‘oyalariga tayanadi [4]: bilim o‘quvchi uchun mazmunli narsani o‘zi yaratish jarayonida eng samarali shakllanadi.

Biroq Scratchdan foydalanishning o‘zi algoritmik tafakkurni rivojlantirmaydi. Muhit ko‘pincha faqat o‘yin yoki animatsiya yaratish vositasi sifatida ishlatiladi; mazmun, ketma-ketlik, faoliyat turlari va natijalarni o‘lchash mezonlari tizimli metodika sifatida ishlab chiqilmasa, o‘quv samarasi tasodifiy bo‘lib qoladi. Aynan shu holat tadqiqotning dolzarbligini belgilaydi.

Tadqiqotning maqsadi – Scratch muhitiga asoslangan, 5–6-sinf o‘quvchilarining yosh xususiyatlariga mos va natijasi o‘lchanadigan algoritmik tafakkurni rivojlantirish metodikasini ilmiy asoslash hamda uning amaliy modelini taklif etishdir.

Tadqiqot vazifalari:

algoritmik tafakkurning mohiyati va tarkibiy qismlarini aniqlash;

Scratch muhitining didaktik imkoniyatlarini va 5–6-sinf o‘quvchilarining yosh xususiyatlarini tahlil qilish;

metodik model, o‘quv modulining mazmuni va dars namunasini ishlab chiqish;

algoritmik tafakkur darajasini baholash rubrikasini yaratish;

pedagogik tajriba-sinov tartibini va natijalarni statistik qayta ishlash usullarini belgilash.

Tadqiqot ob’ekti – umumiy o‘rta ta’lim maktablarining 5–6-sinflarida informatika fanini o‘qitish jarayoni; predmeti – Scratch muhiti vositasida algoritmik tafakkurni rivojlantirish metodikasi.

Tadqiqot farazi: agar o‘qitish “foydalan – o‘zgartir – yarat – anglab yet” modeli bo‘yicha, kompyutersiz faoliyat bilan uyg‘unlashtirilgan loyihaviy topshiriqlar asosida tashkil etilsa va natijalar rubrika yordamida muntazam baholansa, 5–6-sinf o‘quvchilarining algoritmik tafakkur darajasi an’anaviy usulda o‘qitilgan sinflarga nisbatan yuqori bo‘ladi.

Ilmiy yangiligi algoritmik tafakkurning tarkibiy qismlarini Scratch konstruksiyalari va o‘quv faoliyati turlari bilan bog‘lovchi yaxlit metodik modelning taklif etilishida; amaliy ahamiyati esa o‘qituvchilar bevosita qo‘llashi mumkin bo‘lgan modul rejasi, dars ssenariysi va baholash rubrikasining berilishidadir.

Adabiyotlar tahlili va tadqiqot metodologiyasi

1. Algoritmik tafakkurning mohiyati va tarkibi

Hisoblash tafakkuri tushunchasi J. Wing maqolasidan keyin ta’lim tadqiqotlarida keng qo‘llana boshlandi [3]. Uning mazmuni bo‘yicha yagona ta’rif mavjud emas, biroq tadqiqotchilarning ko‘pchiligi abstraksiyalash, masalani qismlarga ajratish, algoritmik fikrlash va nosozliklarni topib tuzatishni uning asosiy elementlari sifatida e’tirof etadi. S. Grover va R. Pea K–12 ta’limidagi tadqiqotlar sharhida bu elementlarni tizimlashtirib, shu bilan birga o‘quvchilar o‘zlashtirishini baholash masalasi dolzarb muammo bo‘lib qolayotganini qayd etadi [5].

K. Brennan va M. Resnick Scratch asosida hisoblash tafakkurini o‘rganish uchun uch o‘lchovli tizim taklif qilgan [6]: (1) hisoblash tushunchalari – ketma-ketlik, takrorlash, parallellik, hodisalar, shartlar, operatorlar va ma’lumotlar; (2) hisoblash amaliyotlari – ishni bosqichma-bosqich olib borish, sinash va nosozliklarni tuzatish, boshqalar ishidan foydalanish va uni qayta ishlash, abstraksiyalash va modullarga ajratish; (3) hisoblash qarashlari – o‘zini ifodalash, boshqalar bilan bog‘lanish va savol berish. Ushbu tizim bizning metodikamizda mazmun va baholash uchun asos bo‘lib xizmat qildi.

Tadqiqot uchun ishchi ta’rif sifatida quyidagisi qabul qilindi: algoritmik tafakkur – masalani qismlarga ajratish, qonuniyatlarni ko‘ra olish, ikkinchi darajali tafsilotlardan chalg‘ib asosiy xususiyatlarni ajratish, yechimni aniq va bajariladigan qadamlar ketma-ketligi shaklida ifodalash hamda uni sinab ko‘rib takomillashtirish qobiliyatidir. Bu ta’rifga ko‘ra, algoritmik tafakkur besh tarkibiy qismdan iborat:

dekompozitsiya – murakkab masalani kichik, yechilishi mumkin bo‘lgan qismlarga bo‘lish;

qonuniyatlarni aniqlash – takrorlanuvchi elementlarni ko‘rish va ulardan takrorlash yoki umumiy qoida shaklida foydalanish;

abstraksiya – ikkinchi darajali tafsilotlardan chalg‘ib, asosiy xususiyatlarni ajratish (masalan, o‘zgaruvchi yoki “Mening bloklarim” yaratish);

algoritm tuzish – yechimni aniq, tartibli, ijrochi bajara oladigan qadamlar ketma-ketligi sifatida ifodalash;

sinash va nosozliklarni tuzatish – natijani kutilgan natija bilan solishtirish, xatoning sababini topish va tuzatish.

2. Scratch muhitining didaktik imkoniyatlari

Scratch yaratuvchilari muhitni “past kirish bo‘sag‘asi, yuqori shift va keng devorlar” (low floor, high ceiling, wide walls) tamoyili asosida loyihalagan [7]: boshlash oson, murakkab loyihalarga o‘sish mumkin va turli yo‘nalishdagi loyihalar yaratish imkoni bor. Dastur bloklarni “yig‘ish” orqali tuziladi, bloklar faqat mantiqan mumkin bo‘lgan tarzda ulanadi, skriptni bosib natijani sahnada darhol ko‘rish mumkin [8]. M. Resnick o‘rganishning to‘rt “P” tamoyilini – loyihalar (Projects), ishtiyoq (Passion), hamkorlik (Peers) va o‘yin (Play) – ta’kidlaydi [9]; bu tamoyillar 5–6-sinf o‘quvchilarining

motivatsiyasi bilan yaxshi mos keladi.

Tahlil natijasida Scratchning algoritmik tafakkurni rivojlantirish nuqtai nazaridan quyidagi didaktik afzalliklari ajratildi:

sintaksis xatolarining deyarli yo‘qligi kognitiv yuklamani algoritmik mantiqqa yo‘naltiradi;

skript natijasi sahnada darhol ko‘rinadi, bu tezkor qayta aloqa beradi va “sinash – tuzatish” siklini tabiiy qiladi;

bir nechta obyektning bir vaqtda ishlashi (parallellik) va hodisalar bilan boshqarish tushunchalarini kundalik tajriba orqali tushuntirish mumkin;

loyihalarni ko‘rish va qayta ishlash (remix) imkoniyati hamkorlik va boshqalar ishini tahlil qilish madaniyatini shakllantiradi;

muhitning oflayn muharriri internet sifati past bo‘lgan maktablarda ham ishlash imkonini beradi.

3. 5–6-sinf o‘quvchilarining yosh xususiyatlari

J. Piaje nazariyasiga ko‘ra, taxminan 11–12 yoshda bola konkret operatsiyalar bosqichidan formal operatsiyalar bosqichiga o‘ta boshlaydi [10]. 5–6-sinf o‘quvchilari (10–12 yosh) aynan shu o‘tish davrida bo‘lib, mavhum tushunchalarni (o‘zgaruvchi, shart, abstraksiya) ko‘z bilan kuzatish mumkin bo‘lgan konkret obrazlarga tayanib o‘zlashtiradi. L. S. Vigotskiyning “yaqin rivojlanish zonasi” tushunchasi [11] esa o‘qituvchi va tengdoshlar yordamida bola mustaqil bajara olmaydigan topshiriqni uddalashi mumkinligini ko‘rsatadi. Bundan metodika uchun quyidagi talablar kelib chiqadi:

mavhum tushunchalarni konkret metaforalar orqali kiritish (masalan, o‘zgaruvchi – nomi yozilgan quti);

juftlikda dasturlash (“haydovchi – navigator” rollari bilan) va kichik guruhlarda ishlash;

topshiriqlarni pog‘onali murakkablashtirish, tayyor namunadan mustaqil loyihaga o‘tish;

dars davomida faoliyat turlarini almashtirib turish, chunki bu yoshdagi o‘quvchilarda barqaror diqqat muddati cheklangan;

o‘yin elementlari va o‘quvchi uchun shaxsiy ma’noga ega bo‘lgan loyiha mavzulari.

4. Tadqiqot usullari va tajriba-sinov dizayni

Tadqiqotda quyidagi usullar qo‘llanadi: nazariy (ilmiy adabiyotlar, davlat ta’lim standartlari va o‘quv dasturlarini tahlil qilish, modellashtirish); empirik (pedagogik kuzatuv, so‘rovnoma, o‘quvchilar loyihalarini rubrika bo‘yicha baholash, algoritmik masalalar bo‘yicha yozma test, Dr. Scratch vositasi yordamida loyihalarni avtomatik tahlil qilish [15]); matematik-statistik usullar.

Pedagogik tajriba-sinov uch bosqichda rejalashtirilgan: aniqlovchi (boshlang‘ich kesim), shakllantiruvchi (metodikani joriy etish) va yakuniy (takroriy kesim va natijalarni qayta ishlash). Tajriba va nazorat guruhlari sifatida bir xil parallel sinflar (har bir guruhda kamida 25–30 nafar o‘quvchi) tanlanadi; nazorat guruhida o‘qitish amaldagi o‘quv dasturi va an’anaviy usullar bo‘yicha, tajriba guruhida esa taklif etilayotgan metodika asosida olib boriladi. Guruhlarning boshlang‘ich bir xilligi dastlabki kesim natijalari bo‘yicha tekshiriladi.

Natijalar va muhokama

1. “Foydalan – o‘zgartir – yarat – anglab yet” metodik modeli

I. Lee va hamkasblari K–12 ta’limida hisoblash tafakkurini rivojlantirish uchun “foydalan – o‘zgartir – yarat” (Use–Modify–Create) ketma-ketligini taklif qilgan [12]. Biz bu modelni to‘rtinchi – refleksiya – bosqichi bilan to‘ldirdik, chunki o‘z faoliyatini anglab yetish hisoblash qarashlarini shakllantirishning zaruriy sharti hisoblanadi [6]. Modelning bosqichlari 1-jadvalda keltirilgan.

1-jadval.

Metodik modelning bosqichlari

Bosqich	O‘qituvchi faoliyati	O‘quvchi faoliyati	Natija
1. Foydalan (Use)	Tayyor loyihani namoyish etadi, yo‘naltiruvchi savollar beradi (“Obyekt nima uchun to‘xtadi?”)	Loyihani ishga tushiradi, skriptni “o‘qiydi”, ishlash mantig‘ini tushuntiradi	Mavjud algoritmni tushunish
2. O‘zgartir (Modify)	Aniq o‘zgartirish topshirig‘ini beradi, ataylab xatoli (“buggy”) skriptlar taqdim etadi	Parametrlarni o‘zgartiradi, xatoni topadi va tuzatadi, yangi imkoniyat qo‘shadi	Algoritm elementlarining rolini tushunish, nosozlikni tuzatish ko‘nikmasi
3. Yarat (Create)	Maslahatchi rolida bo‘ladi, mezonlarni belgilaydi	Yakka yoki juftlikda rejalashtiradi, masalani qismlarga ajratadi, mustaqil loyiha yaratadi	Original loyiha
4. Anglab yet (Reflect)	Muhokama tashkil qiladi, rubrika bo‘yicha baholaydi	Loyihani taqdim etadi, o‘zaro va o‘z-o‘zini baholaydi, “Nimani boshqacha qilgan bo‘lardim?” savoliga javob beradi	Metakognitiv ko‘nikmalar, hisoblash qarashlari

Modelni amalga oshirishda quyidagi tamoyillarga amal qilinadi: (a) har bir yangi mavzu kompyutersiz (unplugged) faoliyat bilan boshlanadi – masalan, sinfda o‘quvchilardan biri “robot”, boshqasi “dasturchi” bo‘lib, buyruqlar ketma-ketligini og‘zaki beradi; (b) topshiriqlar ketma-ketlik, takrorlash, hodisa, shart, o‘zgaruvchi tartibida bosqichma-bosqich murakkablashadi; (c) xato o‘rganish manbai sifatida qaraladi, shu sababli darslarda ataylab xatoli skriptlarni tuzatish topshiriqlari beriladi; (d) loyiha mavzusini tanlashda o‘quvchining shaxsiy qiziqishi hisobga olinadi; (e) har bir mavzu oxirida o‘quvchilar o‘z loyihasini sinfdoshlariga taqdim etadi.

2. O‘quv modulining mazmuni

Algoritmik tafakkurning tarkibiy qismlari va Brennan–Resnick tizimidagi tushunchalar [6] asosida sakkiz mavzudan iborat modul ishlab chiqildi (2-jadval). Modul informatika fanining amaldagi dasturiga qo‘shimcha (variativ) qism sifatida yoki to‘garak mashg‘ulotlarida qo‘llanishi mumkin; soatlar soni o‘quv rejasiga qarab moslashtiriladi.

2-jadval.

Scratch asosidagi o‘quv modulining mazmuni

№	Mavzu	Scratch konstruksiyalari	Tafakkur komponenti	Asosiy faoliyat
1	Algoritm va ijrochi	Obyekt (sprayt), sahna, “Bayroq bosilganda” bloki	Algoritm tuzish	“Robot – dasturchi” o‘yini, birinchi skript
2	Ketma-ketlik	Harakat, tashqi ko‘rinish va tovush bloklari	Algoritm tuzish	“Sayohat” animatsiyasi
3	Takrorlash	“Takrorla”, “Doimiy takrorla” bloklari, qalam (Pen) kengaytmasi	Qonuniyatlarni aniqlash	Geometrik shakllar chizish
4	Hodisalar	Klaviatura va sichqoncha hodisalari, xabar yuborish	Algoritm tuzish, parallellik	Interaktiv tabrik ochiqchasi
5	Shartlar	“Agar … bo‘lsa”, “Agar … aks holda”, sezgi bloklari	Qaror qabul qilish mantig‘i	Labirint o‘yini
6	O‘zgaruvchilar va operatorlar	O‘zgaruvchi, arifmetik va taqqoslash operatorlari, tasodifiy son	Abstraksiya	Matematik viktorina, ball hisoblagich
7	Dekompozitsiya va modullilik	“Mening bloklarim”, obyektlar o‘rtasida xabar almashinuvi	Dekompozitsiya, abstraksiya	Murakkab loyihani qismlarga bo‘lish va rejalashtirish
8	Yakuniy loyiha	Barcha o‘rganilgan konstruksiyalar	Barcha komponentlar, refleksiya	Jamoaviy loyiha va uni taqdim etish

3. Dars namunasi: “Shartlar. Labirint o‘yini” (5-sinf)

Dars maqsadi: o‘quvchilar “agar … bo‘lsa” shart konstruksiyasining ishlash tamoyilini tushunadi, uni takrorlash bilan birga qo‘llab, oddiy o‘yin yaratadi va xatolarni mustaqil topadi. Davomiyligi: 45 daqiqa. Dars bosqichlari:

Motivatsiya (5 daqiqa). Kompyutersiz faoliyat: o‘qituvchi kundalik hayotdan “agar yomg‘ir yog‘sa, soyabon olaman” kabi shartlarni muhokama qildiradi; o‘quvchilar o‘zlari ham shunday shartlar o‘ylab topadi.

Foydalan (7 daqiqa). O‘quvchilar tayyor labirint loyihasini ishga tushiradi va skriptni o‘qib, savollarga javob beradi: “Qahramon devorga tegsa nima bo‘ladi? Buni qaysi blok boshqaradi?”

O‘zgartir (10 daqiqa). Topshiriq: harakat tezligini o‘zgartirish, ikkinchi darajani qo‘shish; shuningdek, o‘qituvchi tayyorlagan xatoli variantni tuzatish (shart bloki takrorlash ichiga qo‘yilmagan, shu sababli o‘yin faqat bir marta tekshiradi).

Yarat (15 daqiqa). Juftliklarda o‘z labirintini yaratish: devor, boshlang‘ich nuqta, finish va “Yutdingiz!” xabari. Mezonlar: kamida ikkita shart, bitta takrorlash, loyiha sinab ko‘rilgan.

Anglab yet (8 daqiqa). Juftliklar loyihalarini almashib sinab ko‘radi, bitta topilgan xato va bitta yaxshi g‘oyani aytadi; o‘z-o‘zini baholash varaqasi to‘ldiriladi.

Labirint o‘yinining algoritmi so‘z bilan quyidagicha ifodalanadi: (1) “Bayroq bosilganda” qahramon boshlang‘ich nuqtaga o‘tadi; (2) “Doimiy takrorla” ichida: agar o‘ng strelka bosilgan bo‘lsa – x koordinata 5 ga oshadi, chap strelka bo‘lsa – 5 ga kamayadi, yuqori va pastki strelkalar uchun y koordinata xuddi shunday o‘zgaradi; (3) agar qahramon devor rangiga tegsa – boshlang‘ich nuqtaga qaytadi; (4) agar finish rangiga tegsa – “Yutdingiz!” deb aytadi va barcha skriptlar to‘xtatiladi. Shu darsning o‘zida dekompozitsiya (harakat, devor bilan to‘qnashuv, g‘alaba shartlari), takrorlash va shart tushunchalari bir butun holda namoyon bo‘ladi.

6-sinf uchun mos loyiha namunasi – “Matematik viktorina”: dastur tasodifiy sonlar tanlab misol beradi (operatorlar, tasodifiy son), foydalanuvchi javobini kiritadi (“So‘ra va kut” bloki), javob to‘g‘ri bo‘lsa “Ball” o‘zgaruvchisi bittaga oshadi, noto‘g‘ri bo‘lsa kamayadi yoki o‘zgarmaydi (shart). Bunday loyiha o‘zgaruvchi tushunchasini, ma’lumot almashinuvini va sinash zaruratini tabiiy ravishda kiritadi hamda fanlararo bog‘lanishni (matematika) ta’minlaydi.

Amaliyotda eng ko‘p uchraydigan qiyinchiliklar: shart blokini takrorlash ichiga joylashtirmaslik (shart bir marta tekshirilib, dastur tugaydi); koordinatalar tizimini tushunmaslik; o‘zgaruvchining boshlang‘ich qiymatini belgilamaslik (loyiha qayta ishga tushirilganda eski ball saqlanib qoladi); ko‘p skriptlarning bir vaqtda ishlashini inobatga olmaslik. Shuning uchun darslarda bunday xatolarni ataylab o‘z ichiga olgan topshiriqlar beriladi.

4. Algoritmik tafakkur darajasini baholash

Algoritmik tafakkurni faqat yakuniy loyihaning “chiroyli” chiqqanligi bo‘yicha baholab bo‘lmaydi. Shu sababli baholash ko‘p manbali tarzda tashkil etiladi: (1) o‘quvchi loyihalarini rubrika bo‘yicha baholash (3-jadval); (2) Dr. Scratch kabi vositalar bilan loyiha murakkabligini avtomatik tahlil qilish [15]; (3) xalqaro Bebras tanlovi uslubidagi qisqa algoritmik masalalardan iborat yozma test; (4) loyiha ustida ishlash jarayonini kuzatish va o‘z-o‘zini baholash varaqalari. Test va topshiriqlarni tuzishda Blum taksonomiyasining L. Anderson va D. Krathwohl tomonidan yangilangan variantidagi sathlar (eslash, tushunish, qo‘llash, tahlil, baholash, yaratish) hisobga olinadi [16].

3-jadval.

Algoritmik tafakkurni baholash rubrikasi (har bir mezon 1–4 ball)

Mezon	1 – boshlang‘ich	2 – o‘rta	3 – yetarli	4 – yuqori
Ketma-ketlik va takrorlash	Bloklar tartibsiz, takrorlash ishlatilmaydi yoki noto‘g‘ri qo‘llanadi	Oddiy ketma-ketlik to‘g‘ri, takrorlash faqat o‘qituvchi yordamida	Ketma-ketlik va takrorlash mustaqil va to‘g‘ri qo‘llanadi	Takrorlashdan samarali foydalanib, ortiqcha bloklarni qisqartiradi
Shartlar va hodisalar	Hodisa yoki shart ishlatilmaydi	Bitta oddiy hodisa yoki shart, yordam bilan	Bir nechta hodisa va “agar – aks holda” mustaqil to‘g‘ri qo‘llanadi	Ichma-ich shartlar va obyektlar o‘rtasida xabar almashinuvini qo‘llaydi
O‘zgaruvchilar va ma’lumotlar	Ishlatilmaydi	Tayyor o‘zgaruvchidan foydalanadi, o‘zi yarata olmaydi	O‘zgaruvchi yaratadi va qiymatini to‘g‘ri yangilaydi	O‘zgaruvchi va operatorlar yordamida loyiha mantig‘ini boshqaradi
Dekompozitsiya va modullilik	Loyiha reja va tuzilmasiz	Reja bor, qismlar o‘zaro bog‘lanmagan	Loyiha mantiqiy qismlarga (obyektlar, sahnalar) bo‘lingan	Takrorlanuvchi qismlar alohida blok sifatida ajratilgan, reja to‘liq bajarilgan
Sinash va nosozliklarni tuzatish	Xatoni sezmaydi	Xatoni sezadi, tuzatishda yordam so‘raydi	Dasturni sinaydi, xato sababini topib mustaqil tuzatadi	Turli holatlarda sinaydi, xatolarni tizimli tuzatadi va boshqalarga tushuntiradi

Jami ball 5 dan 20 gacha oraliqda bo‘lib, umumiy daraja quyidagicha belgilanadi: 5–8 ball – boshlang‘ich, 9–12 ball – o‘rta, 13–16 ball – yetarli, 17–20 ball – yuqori daraja.

5. Muhokama. Taklif etilgan metodikaning ilmiy asoslari xalqaro tadqiqotlar natijalari bilan mos keladi. Lye va Koh K–12 ta’limida dasturlash orqali hisoblash tafakkurini o‘rgatishga oid tadqiqotlarni tahlil qilib, ko‘pchilik ishlar dasturlash tushunchalariga e’tibor qaratganini, amaliyotlar va qarashlar esa kamroq o‘rganilganini ko‘rsatgan [13]. Bizning modelimiz va rubrikamiz bu bo‘shliqni to‘ldirishga qaratilgan: “O‘zgartir” va “Anglab yet” bosqichlari hamda “Sinash va nosozliklarni tuzatish”, “Dekompozitsiya va modullilik” mezonlari amaliyot va qarashlarni bevosita o‘lchaydi. Scratchdan foydalanishga oid tizimli sharhlar muhitning ijobiy imkoniyatlarini ko‘rsatadi, shu bilan birga natija o‘qitish usuli va o‘qituvchi yordamiga bog‘liqligini qayd etadi [13; 14]. Bu esa muhitning o‘zi emas, aynan tizimli metodika hal qiluvchi ahamiyatga ega ekanini tasdiqlaydi.

Metodikani joriy etishda quyidagi xavflarni hisobga olish zarur. Birinchidan, loyiha “o‘yin” darajasida qolib, algoritmik mazmun yo‘qolishi mumkin; buning oldini rubrika va aniq mezonlar oladi. Ikkinchidan, o‘quvchilar bir-biridan tayyor skriptni ko‘chirib olishi mumkin; shu sababli “O‘zgartir” va “Anglab yet” bosqichlarida o‘quvchi o‘z qarorini og‘zaki izohlashi talab qilinadi. Uchinchidan, tajriba natijasini faqat metodikaga bog‘lash uchun nazorat guruhi bilan taqqoslash va guruhlar boshlang‘ich bir xilligini tekshirish shart. To‘rtinchidan, o‘qituvchining tayyorgarligi va kompyuter jihozlari yetarliligi samaradorlikka bevosita ta’sir qiladi; resurslar cheklangan sharoitda juftlikda ishlash va oflayn muharrirdan foydalanish amaliy yechim bo‘lishi mumkin.

Informatika o‘qituvchilari uchun amaliy tavsiyalar:

har bir yangi tushunchani kompyutersiz faoliyat va kundalik hayotdan misol bilan boshlash;

darsning kamida uchdan bir qismini o‘quvchilarning mustaqil yoki juftlikdagi ijodiy ishiga ajratish;

xatoli skriptlarni tuzatish topshiriqlarini muntazam berish;

baholashda tayyor loyihaning tashqi ko‘rinishi emas, algoritm mantig‘i va sinash jarayoniga e’tibor qaratish;

o‘quvchilar loyihalarini taqdim etish va muhokama qilish uchun sinf ichida “loyihalar ko‘rgazmasi” tashkil etish.

Xulosa. Olib borilgan tahlil va ishlab chiqilgan model quyidagi xulosalarga asos bo‘ladi:

Algoritmik tafakkur dekompozitsiya, qonuniyatlarni aniqlash, abstraksiya, algoritm tuzish va nosozliklarni tuzatishdan iborat yaxlit qobiliyat bo‘lib, uni Scratch konstruksiyalari va faoliyat turlari bilan bog‘lab rivojlantirish mumkin.

Scratch muhiti sintaksis to‘siqlarining yo‘qligi, tezkor vizual qayta aloqasi va loyihaviy xususiyati tufayli 5–6-sinf o‘quvchilarining yosh xususiyatlariga mos keladi, biroq samara faqat tizimli metodika bilan birgalikda yuzaga chiqadi.

“Foydalan – o‘zgartir – yarat – anglab yet” modeli, sakkiz mavzudan iborat modul va dars ssenariylari o‘qituvchiga algoritmik tafakkurni bosqichma-bosqich rivojlantirish imkonini beradi.

Besh mezonli rubrika, avtomatik tahlil vositasi va yozma test algoritmik tafakkurni ko‘p manbali va o‘lchanadigan tarzda baholashni ta’minlaydi.

Taklif etilgan metodikaning samaradorligi tajriba va nazorat guruhlari bilan o‘tkaziladigan pedagogik tajriba-sinovda, statistik mezonlar va effekt hajmi yordamida tasdiqlanishi lozim; kelgusi tadqiqotlarni uzoq muddatli kuzatuv va metodikani boshqa fanlarda qo‘llash yo‘nalishida davom ettirish maqsadga muvofiq.

Foydalanilgan adabiyotlar ro'yxati

[1] O‘zbekiston Respublikasining “Ta’lim to‘g‘risida”gi Qonuni, 2020-yil 23-sentabr, O‘RQ-637-son.

[2] O‘zbekiston Respublikasi Prezidentining “Raqamli O‘zbekiston – 2030” strategiyasini tasdiqlash va uni samarali amalga oshirish chora-tadbirlari to‘g‘risida”gi 2020-yil 5-oktabrdagi PF-6079-son Farmoni.

[3] Wing J. M. Computational thinking // Communications of the ACM. – 2006. – Vol. 49, No. 3. – P. 33–35.

[4] Papert S. Mindstorms: Children, Computers, and Powerful Ideas. – New York: Basic Books, 1980. – 230 p.

[5] Grover S., Pea R. Computational thinking in K–12: A review of the state of the field // Educational Researcher. – 2013. – Vol. 42, No. 1. – P. 38–43.

[6] Brennan K., Resnick M. New frameworks for studying and assessing the development of computational thinking // Proceedings of the 2012 Annual Meeting of the American Educational Research Association. – Vancouver, Canada, 2012. – 25 p.

[7] Resnick M., Maloney J., Monroy-Hernández A. et al. Scratch: Programming for all // Communications of the ACM. – 2009. – Vol. 52, No. 11. – P. 60–67.

[8] Maloney J., Resnick M., Rusk N., Silverman B., Eastmond E. The Scratch programming language and environment // ACM Transactions on Computing Education. – 2010. – Vol. 10, No. 4. – Article 16.

[9] Resnick M. Lifelong Kindergarten: Cultivating Creativity through Projects, Passion, Peers, and Play. – Cambridge, MA: MIT Press, 2017. – 216 p.

[10] Piaget J., Inhelder B. The Psychology of the Child. – New York: Basic Books, 1969. – 173 p.

[11] Vygotsky L. S. Mind in Society: The Development of Higher Psychological Processes. – Cambridge, MA: Harvard University Press, 1978. – 159 p.

[12] Lee I., Martin F., Denner J. et al. Computational thinking for youth in practice // ACM Inroads. – 2011. – Vol. 2, No. 1. – P. 32–37.

[13] Lye S. Y., Koh J. H. L. Review on teaching and learning of computational thinking through programming: What is next for K-12? // Computers in Human Behavior. – 2014. – Vol. 41. – P. 51–61.

[14] Zhang L., Nouri J. A systematic review of learning computational thinking through Scratch in K-9 // Computers & Education. – 2019. – Vol. 141. – Article 103607.

[15] Moreno-León J., Robles G., Román-González M. Dr. Scratch: Automatic analysis of Scratch projects to assess and foster computational thinking // RED. Revista de Educación a Distancia. – 2015. – No. 46. – P. 1–23.

[16] Anderson L. W., Krathwohl D. R. (eds.). A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives. – New York: Longman, 2001. – 352 p.

[17] Scratch Foundation. Scratch – Imagine, Program, Share [Electronic resource]. – URL: https://scratch.mit.edu (kirilgan sana: 06.11.2026).