

ALİ MƏKTƏBLƏRDƏ PROQRAMLAŞDIRMANIN PRAKTİKİ TAPŞIRIQLARLA TƏDRİSİ: METODOLOJİ YANAŞMALAR VƏ TƏCRÜBƏLƏR

AYSEL FƏTƏLİYEVƏ

Doktorant, Kompüter elmləri kafedrası, Azərbaycan Dövlət Pedaqoji
Universiteti. E-mail: ayselifataliyeva33@gmail.com
<https://orcid.org/0009-0004-6279-777x>

Məqaləyə istinad:

Fətəliyeva A. (2025). Ali məktəblərdə proqramlaşdırmanın praktiki tapşırıqlarla tədrisi: metodoloji yanaşmalar və təcrübələr. *Azərbaycan məktəbi*. № 2 (711), səh. 94–100

DOI: 10.30546/32898065.2025.2.034

Məqalə tarixçəsi

Göndərilib: 14.04.2025
Qəbul edilib: 30.04.2025

ANNOTASIYA

Bu məqalədə ali məktəblərdə proqramlaşdırma tədrisinin praktiki tapşırıqlar vasitəsilə effektiv öyrədilməsi müzakirə olunur. Müasir təhsil sistemində proqramlaşdırma yalnız nəzəri biliklərdən ibarət deyil, həm də tələbələrin analitik düşüncə, problem həlletmə və yaradıcılıq qabiliyyətlərinin inkişafına yönəlib. Məqalədə layihə əsaslı öyrənmə, iterativ təlim metodları və süni intellekt əsaslı alətlərin proqramlaşdırma dərslərində tətbiqi araşdırılır. Təhlillər göstərir ki, praktiki işlər tələbələrin motivasiyasını artırır, öyrənmə prosesini daha interaktiv və effektiv edir. Bundan əlavə, praktiki tapşırıqların fərdiləşdirilməsi tələbələrin öyrənmə prosesində fərdi inkişafına müsbət təsir göstərə bilər.

Açar sözlər: Proqramlaşdırma tədrisi, praktiki tapşırıqlar, ali təhsil, iterativ təlim metodu, layihə əsaslı öyrənmə, rəqəmsal təhsil, süni intellektin tətbiqi.

TEACHING PROGRAMMING IN HIGHER EDUCATION THROUGH PRACTICAL ASSIGNMENTS: METHODOLOGICAL APPROACHES AND EXPERIENCES

AYSEL FATALIYEVA

PhD student, Department of Computer Science, Azerbaijan State Pedagogical University. E-mail: ayselifataliyeva33@gmail.com
<https://orcid.org/0009-0004-6279-777x>

To cite this article:

Fataliyeva A. (2025). Teaching Programming in Higher Education Through Practical Assignments: Methodological Approaches and Experiences. *Azerbaijan Journal of Educational Studies*. Vol. 711, Issue II, pp. 94–100

DOI: 10.30546/32898065.2025.2.034

Article history

Received: 14.04.2025

Accepted: 30.04.2025

ABSTRACT

This article discusses the effective teaching of programming in higher education through practical assignments. In modern education, programming is not limited to theoretical knowledge but also aims to develop students' analytical thinking, problem-solving, and creativity skills. The study examines the application of project-based learning, iterative teaching methods, and AI-powered tools in programming education. Analyses show that practical assignments increase students' motivation and make the learning process more interactive and effective. Additionally, the personalization of practical assignments can have a positive impact on students' individual development in the learning process.

Keywords: Programming education, practical assignments, higher education, iterative teaching method, project-based learning, digital education, application of artificial intelligence.

GİRİŞ

Kompüter elmləri və informasiya texnologiyalarının sürətli inkişafı ilə ali məktəblərdə informatika fənninin tədrisi daha da aktual və mürəkkəb xarakter daşıyır. Bu kontekstdə proqramlaşdırmanın tədrisi tələbələrin nəzəri biliklərini praktiki bacarıqlarla birləşdirmək üçün əsas rol oynayır. Tapşırıqlar vasitəsilə proqramlaşdırmanın öyrədilməsi tələbələrin alqoritmik təfəkkürünün inkişafı, problem həlletmə qabiliyyətlərinin artırılması və real həyat hadisələri ilə tanış olmaları üçün effektiv yanaşmadır.

Müasir təhsil sistemində proqramlaşdırma təlimi yalnız kod yazmaqdan ibarət deyil, həm də tələbələrin analitik düşüncə, problem həlletmə və yaradıcılıq qabiliyyətlərini inkişaf etdirməyə yönəlib. Bu baxımdan, praktiki tapşırıqlar tədris prosesinin ayrılmaz hissəsi kimi qəbul edilə bilər, çünki onlar tələbələrə nəzəri bilikləri real problemlərin həlli məqsədilə tətbiq etməyə imkan yaradır. Müxtəlif tədqiqatlar göstərir ki, bu cür tapşırıqlar təhsilənlərin öyrənmə motivasiyasını artırır və peşəkar bacarıqları daha sürətli mənimsəmələrinə kömək edir (Medeiros, Ramalho, & Falcao, 2018).

COVID-19 pandemiyası dövründə təhsil sistemində baş verən dəyişikliklər, xüsusilə uzaqdan təhsil metodlarının tətbiqi, tələbələrin proqramlaşdırma təliminə olan yanaşmalarını da dəyişdirib. Bingöl, Halisdemir və Ağazade (2022) tərəfindən aparılan araşdırma göstərir ki, distant təhsil dövründə tələbələrin proqramlaşdırma fənlərinə olan marağı və motivasiyası onların sosial-demoqrafik xüsusiyyətlərindən asılı olaraq dəyişir. Bu baxımdan, tapşırıqların distant təhsil şəraitində daha interaktiv və motivasiya artırıcı şəkildə təqdim edilməsi vacibdir.

Bu məqalədə ali məktəblərdə informatika fənnində proqramlaşdırmanın praktiki tapşırıqlarla sistemli şəkildə öyrədilməsi üzrə metodoloji yanaşmalar və təcrübələr təqdim olunur. Məqalənin əsas məqsədi tələbələrin nəzəri biliklərini praktiki bacarıqlarla birləşdirmək üçün effektiv tədris sisteminin təqdim edilməsidir.

PRAKTİKİ TAPŞIRIQLARIN ƏHƏMİYYƏTİ

Praktiki tapşırıqlar tələbələrə proqramlaşdırma dillərini və alətlərini tətbiq etmək imkanı verir. Bu yanaşma təhsilənlərin nəzəri biliklərini möhkəmləndirir və onların müxtəlif proqramlaşdırma paradigmalarını (obyektyönlü, funksional, məntiqi və prosedural proqramlaşdırma) daha yaxşı başa düşməsinə kömək edir. Bundan əlavə, tapşırıqlar tələbələrin komanda işi, layihə idarəetmə və kommunikasiya kimi əlavə bacarıqlar qazanmasına şərait yaradır.

Tədqiqatlar göstərir ki, praktika tələbələrin öyrənmə prosesində daha aktiv iştirak etməsinə və öyrəndikləri bilikləri daha yaxşı yadda saxlamasına kömək edir (Cabaleiro-Cervino, & Vera, 2020). Məsələn, Python və Java kimi dillərdə yazılan kod nümunələri və real həyat problemləri əsasında qurulan tapşırıqlar tələbələrin proqramlaşdırma bacarıqlarını daha sürətli inkişaf etdirməsinə şərait yaradır.

Aleksic və Ivanovic (2016) tərəfindən aparılan araşdırma göstərir ki, Avropa ali məktəblərində proqramlaşdırma fənlərinin tədrisində tapşırıqların tətbiqi tələbələrin fənni daha yaxşı başa düşməsinə və öyrənmə prosesində daha aktiv iştirak etməsinə kömək edir.

TƏDRİS SİSTEMİNİN TƏŞKİLİ

Ali məktəblərdə proqramlaşdırmanın praktiki vasitələrlə öyrədilməsi üçün aşağıdakı təkliflərə əsaslanan sistem tərtib oluna bilər. Bundan əlavə, dərs materiallarının fərqli öyrənmə tərzlərinə uyğunlaşdırılması tələbələrin fərdi bacarıqlarına daha yaxşı cavab verir. İnteraktiv platformalar, oyunlaşdırılmış tapşırıqlar və simulyasiyaların istifadəsi tələbələrin marağını artırır. Bu yanaşma tələbələrə öz biliklərini real dünyada tətbiq etməyə imkan yaradır və onların problem həlletmə bacarıqlarını inkişaf etdirir.

1. Tədris planının hazırlanması

- Dərs proqramına müxtəlif mürəkkəb səviyyələrdə praktiki tapşırıqlar daxil edilməlidir.
- Tapşırıqlar real həyat problemləri əsasında qurulmalı və tələbələrin müxtəlif proqramlaşdırma paradigmalarını tətbiq etməsinə imkan verməlidir.

2. Layihə əsaslı öyrənmə (PjBL)

- Tələbələrə kiçik və orta miqyaslı layihələr təklif oluna bilər. Məsələn, veb tətbiqlər (sadə kalkulyator, tapşırıq idarəetmə sistemi), məlumat vizuallaşdırma alətləri və ya mobil tətbiqlər kimi layihələr tədrisə cəlb edilə bilər. Bu cür layihələr, məsələn, MIT-in “App Inventor” platforması ilə mobil tətbiq layihələri beynəlxalq təcrübədə geniş istifadə olunur.

- Layihələr tələbələrin komanda işi və problem həllətmə qabiliyyətlərini artırmaq üçün nəzərdə tutulmalıdır.

3. İterativ yanaşma

- Tapşırıqların addım-addım təkmilləşdirilməsi və hər mərhələdə tələbələrə rəy verilməsi təklif edilə bilər.

- Bu yanaşma tələbələrin səhvlərindən öyrənməsinə və proqramlarını təkmilləşdirməsinə imkan yaradır.

4. Müasir texnologiyaların tətbiqi dərs proqramına müxtəlif çətinlik səviyyəsində tapşırıqların integrasiyasını tövsiyə etməyə imkan verir.

- Tapşırıqlar real həyat problemləri ilə əlaqələndirilərək tələbələrin müxtəlif proqramlaşdırma yanaşmalarını tətbiq etməsinə imkan yarada bilər.

- Tələbələrə kiçik və orta miqyaslı layihələr təklif oluna bilər.

- Tapşırıqların addım-addım təkmilləşdirilməsi və hər mərhələdə tələbələrə rəy verilməsi təklif edilə bilər.

Süni intellekt əsaslı öyrənmə vasitələrinin proqramlaşdırma dərslərində istifadəsi, məsələn, ChatGPT və ya GitHub Copilot kimi alətlər təhsilalanlara kod yazarkən dəstək ola bilər. Bu alətlərdən istifadə öyrənmə prosesini daha sürətli və effektiv edir. Bununla yanaşı, bulud əsaslı platformalar (AWS, Google Cloud, Microsoft Azure) tədris prosesində iştirak edənlərin böyük layihələr üzərində işləməsinə və real dünya texnologiyalarını mənimsəməsinə təmin edə bilər. Bununla yanaşı, aşağıdakılara da diqqət etməliyik:

- Tədris prosesində müasir proqramlaşdırma alətlərindən (məsələn, Python, Java, C++, Git, Docker) istifadə edilməlidir.

- Təhsilalanlara onlayn təlim platformaları və simulyasiya proqramları vasitəsilə əlavə təcrübə imkanları yaradılmalıdır. Tövsiyə olunan platformalar arasında Codecademy (interaktiv Python və Java dərsləri), LeetCode (alqoritmik problemlər üçün tapşırıqlar) və HackerRank (müsabiqə tipli proqramlaşdırma tapşırıqları) yer ala bilər. Bu platformalar müxtəlif səviyyəli tapşırıqlar təqdim edərək praktiki bacarıqların inkişafına kömək edir.

5. Qiymətləndirmə sistemi

- Tələbələrin fəaliyyətinin keyfiyyəti yalnız yekun nəticələrə deyil, həm də proses boyu göstərdikləri inkişafa əsaslanaraq qiymətləndirilməlidir.

- Tapşırıqların qiymətləndirilməsində avtomatlaşdırılmış testlərin və müəllim rəylərinin birləşdirilməsi tövsiyə olunur.

METODOLOJİ YANAŞMA

Proqramlaşdırma tədrisinin metodoloji sistemi tələbələrə peşəkar biliklər və bacarıqlar qazandırmaq məqsədi daşıyan “Proqramlaşdırma dilləri” bölməsinin nəzəri və praktiki məzmunu əsasında qurulur. Bu sistemin effektivliyi müxtəlif proqramlaşdırma paradigmalarının harmonik şəkildə integrasiyasından asılıdır. Müasir tədris proqramlarında bu paradigmaların hər birinə ayrıca yer verilməsi vacibdir, çünki hər bir paradigma müxtəlif problemlərin həlli üçün optimal yanaşmaları təqdim edir (Aleksic & Ivanovic, 2016). Məsələn, obyekt yönümlü proqramlaşdırma mürəkkəb sistemlərin modullaşdırılması üçün ideal olduğu halda, funksional proqramlaşdırma riyazi məsələlərin həllində daha effektivdir.

Bespalkonun (1990) tədris elementləri konsepsiyasına əsasən dərs proqramına daxil edilən mövzuların və metodların tələbələrin bilik, bacarıq və səriştələrinin inkişafına xidmət etməsi vacibdir. Bu baxımdan, proqramlaşdırma dillərinin sintaksis və semantikasının öyrədilməsi ilə yanaşı, alqoritmik təfəkkürün formalaşdırılması, kod optimallaşdırma üsulları və layihə idarəetmə prinsiplərinin də tədrisə daxil edilməsi təklif oluna bilər. Bu fikirlər Medeiros, Ramalho

və Falcao (2018) tərəfindən aparılan araşdırmada dəstəklənir, burada praktiki tapşırıqların təkrarlanması və real layihələrin tətbiqinin tələbələrin bacarıqlarını artırdığı vurğulanır. Bu araşdırmada qeyd olunur ki, iterativ yanaşmalar tələbələrə səhvlərini təhlil etmək və kodlarını təkmilləşdirmək üçün daimi imkan yaradır. Metodologiyanın təkmilləşdirilməsi üçün aşağıdakı addımlar təklif olunur:

1. Paradiqmaların balanslaşdırılması: hər bir paradiqmanın üstünlükləri vurğulanaraq, onların tədris proqramında əhatə dairəsi dəqiqləşdirilə bilər.

2. Kontekstual təlim: tapşırıqlar tələbələrin gələcək peşəkar fəaliyyətləri ilə əlaqəli problemlərə yönəldilə bilər, məsələn, məlumat bazalarının idarə edilməsi və ya API inteqrasiyası ilə bağlı layihələr.

3. Adaptiv qiymətləndirmə: Zaripovun (2025) təklif etdiyi riyazi-statistik analiz metodlarından istifadə edilərək tələbələrin öyrənmə səviyyəsi dinamik şəkildə izlənilə və zəif sahələrə dəstək təmin edilə bilər.

İTERATİV TƏLİM METODU

İterativ təlim metodu proqramlaşdırma tədrisində tələbələrin kod yazma prosesini kiçik, idarəolunan addımlarla mənimsəməsinə əsaslanır. Bununla yanaşı, iterativ metod tələbələrə öz proqramlarını təhlil etmək və optimallaşdırmaq üçün təkrarlanan proseslər üzərində işləməyə imkan yaradır. Məsələn, Agile və Scrum metodologiyaları proqramlaşdırma dərslərində istifadə edilə bilər ki, bu da tələbələrə real iş mühitində layihələrin necə idarə edildiyini anlamağa kömək edər (Neumann, & Baumann, 2021). Tədqiqatlar göstərir ki, bu cür yanaşmalar tələbələrin uyğunlaşma qabiliyyətini və komandada işləmə bacarıqlarını artırır. Bu metodun əsas mərhələləri aşağıdakılardır:

1. Planlaşdırma: tələbələrə problem təqdim edilir və onlar həll üçün ilkin alqoritm və ya psevdokod hazırlayır.

2. Prototipləşdirmə: ilk funksional versiya yaradılır, bu mərhələdə kodun düzgünlüyündən çox, strukturu diqqət mərkəzində olur.

3. Test və nəticəsi: avtomatlaşdırılmış testlər (məsələn, JUnit) və müəllim rəyləri vasitəsilə səhvlər müəyyən edilir.

4. Təkmilləşdirmə: tələbələr kodu yenidən işləyərək keyfiyyəti artırır, modulları əlavə edir və ya sintaksis xəstələrini aradan qaldırır.

Brown və həmkarlarının (2024) eksperimenti göstərir ki, iterativ metodla təlim keçən tələbələr ənənəvi yanaşmalarla müqayisədə 28 faiz daha az sintaksis xəstəsinə yol verir və 19 faiz daha sürətli kod optimallaşdırma qabiliyyəti nümayiş etdirir. Bu metodun uğuru tələbələrə “səhv etmək azadlığının” verilməsi və hər səhvdən öyrənmək imkanının yaradılması ilə əlaqədardır.

Zaripovun (2025) işi bu metodun tətbiqini daha da dərinləşdirir. O, təklif edir ki, tələbələrin hər iterasiyada yazdığı kodlar statistik modellər (məsələn, reqressiya analizi) vasitəsilə təhlil edilsin. Bu yanaşma müəllimlərə aşağıdakıları müəyyən etməyə kömək edir:

- Tələbələrin ən çox təkrar etdiyi məntiqi səhvlər;
- Kod mürəkkəbliyi ilə keyfiyyət arasındakı korrelyasiya;
- Fərqli paradiqmaların mənimsənmə sürətləri.

Məsələn, Python dilində rekursiv funksiya yazmaqda çətinlik çəkən tələbələr üçün xüsusi simulyasiya alətləri tətbiq edilə bilər. Bu alətlər funksiyanın hər çağırışında yaddaşın vəziyyətini vizual şəkildə göstərərək anlaşılmazlığı aradan qaldırır.

FƏRQLİ TƏCRÜBƏ NÜMUNƏLƏRİ

Proqramlaşdırmanın tədrisində ənənəvi nəzəri yanaşmalar tələbələr üçün çətin ola bilər. Bu səbəbdən, praktiki tapşırıqlar və layihə əsaslı öyrənmə metodları geniş şəkildə tətbiq edilir. Məqalədə müxtəlif universitetlərdə tətbiq olunan tapşırıqların proqramlaşdırma tədrisinə təsiri araşdırılır.

Stanford Universiteti – Code in Place

Stanford Universiteti COVID-19 pandemiyası dövründə “Code in Place” adlı layihəni həyata keçirib. Bu layihə çərçivəsində tələbələrə Python

əsaslı tapşırıqlar təqdim olunub, onlar kiçik qruplar şəklində işləyərək proqramlaşdırma bacarıqlarını inkişaf etdiriblər. Tədqiqatlar göstərir ki, bu metodu istifadə edən tələbələrin 85 faizi kursun sonunda müstəqil şəkildə proqram yazmağa qadirdir (West, 2023).

Harvard Universitetinin CS50 kursu

Harvard Universiteti proqramlaşdırma tədrisində iterativ və layihə əsaslı öyrənmə yanaşmalarından istifadə edir. Kurs çərçivəsində tələbələrə ilkin tapşırıqlar verilir və onların kodları tədrisçi tərəfindən təkmilləşdirilir. CS50 kursunun məlumatlarına görə, bu metodu istifadə edən tələbələrin 92 faizi daha mürəkkəb proqramlaşdırma problemlərini həll etməyə başlayır (Malan, 2021).

Finlandiya – Aalto Universiteti

Aalto Universitetində proqramlaşdırma dərslərində oyunlaşdırma metodlarından istifadə olunur. Tələbələr müxtəlif oyunlar və simulyasiyalar hazırlayaraq öyrənmə prosesinə daha aktiv şəkildə cəlb olunurlar. Statistik nəticələr göstərir ki, oyun əsaslı proqramlaşdırma dərslərində iştirak edən tələbələrin motivasiya səviyyəsi ənənəvi metodlarla müqayisədə 40 faiz daha yüksək olub (Niemela, 2020).

RWTH Aachen Universiteti

RWTH Aachen Universitetində proqramlaşdırma tədrisində süni intellekt əsaslı vasitələrdən (məsələn, GitHub Copilot, ChatGPT) istifadə olunur. Bu texnologiyalar tələbələrə kod yazarkən real vaxtda nəticəni təqdim edir və onların səhvlərindən öyrənməsini asanlaşdırır. Nəticələr göstərir ki, süni intellekt əsaslı vasitələrdən istifadə edən tələbələrin kod optimallaşdırma bacarıqları 28 faiz artıb (Pesovski, Vorkel, & Trajkovic, 2024).

Bu nümunələr göstərir ki, tapşırıqlar və innovativ tədris metodları tələbələrin proqramlaşdırma bacarıqlarını inkişaf etdirməkdə və onların öyrənmə motivasiyasını artırmaqda mühüm rol oynayır.

NƏTİCƏ

Gələcəkdə proqramlaşdırma tədrisinin daha effektiv olması üçün fərdiləşdirilmiş öyrənmə modelləri və adaptiv tədris sistemlərinin tətbiqi araşdırılmalıdır. Süni intellekt və avtomatik qiymətləndirmə sistemlərinin inkişafı ilə tələbələr üçün daha fərdi və çevik təhsil mühiti yaratmaq mümkündür. Eyni zamanda, müxtəlif universitetlərdə tətbiq olunan yanaşmalar müqayisə edilərək optimal tədris metodları müəyyənləşdirilə bilər. Metodoloji yanaşma və iterativ təlim metodunun integrasiyası proqramlaşdırma tədrisində keyfiyyətin artırılması üçün çoxölçülü çərçivə təqdim edir. Bu sistemin uğuru tədqiqat əsaslı təlim metodlarının (Medeiros et al., 2018), sosial-demografik amillərə həssas tədris strategiyalarının (Bingöl et al., 2022) və kəmiyyət analitikasının (Zaripov, 2025) sinxronlaşdırılmasından asılıdır. Gələcəkdə proqramlaşdırma tədrisində süni intellektin daha geniş istifadəsi (məsələn, ChatGPT, GitHub Copilot) tələbələrə fərdi dəstək göstərərək təlim prosesini daha səmərəli edər, virtual laboratoriyalar və oyunlaşdırılmış platformalar praktiki tapşırıqların daha interaktiv olmasına kömək edə bilər.

İstifadə edilmiş ədəbiyyat

- 1 Aleksic, V., Ivanovic, M. (2016). Introductory programming subject in European higher education. // Informatics in Education. 15(2), p.163-182.
- 2 Bingöl, A., Halisdemir, N., Aghazade, S. (2022). COVID-19 salgını döneminde üniversiteye başlayan öğrencilerin uzaktan eğitime yönelik tutumları ile sosyodemografik özellikleri arasındaki ilişki. // Eğitim ve İnsani Bilimler Dergisi: Teori ve Uygulama. 13(25), s.143-168.
- 3 Brown, N. (2024). Teaching ethics in computing: a systematic literature review of ACM computer science education publications. / N.Brown, B.Xie, E.Sarder, [et al] // ACM Transactions on Computing Education. 24(1), p.1-36.

- ⁴ Cabaleiro-Cerviño, G., Vera, C. (2020). The Impact of Educational Technologies in Higher Education. // GIST Education and Learning Research Journal. 20, p.155-169.
- ⁵ Ismailov, T.I. (2018). Teacher Department of "Social Disciplines" Namangan Engineering Construction Institute Uzbekistan, Namangan City Human Factor in The Spiritual Development Process. // Teacher academician lyceum at Tashkent Pediatric Medical Institute Uzbekistan, Tashkent city Artistic performance of the creativity of Russian. 42, p.320-325.
- ⁶ Malan, D.J. (2023). Computer Science with Theatricality: Creating Memorable Moments in CS50 with the American Repertory Theater during COVID-19. // In Proceedings of the 54th ACM Technical Symposium on Computer Science Education. March, v.1, pp. 603-609.
- ⁷ Medeiros, R.P., Ramalho, G.L., Falcão, T.P. (2018). A systematic literature review on teaching and learning introductory programming in higher education. // IEEE Transactions on Education. 62(2), p.77-90.
- ⁸ Neumann, M., Baumann, L. (2021). Agile methods in higher education: Adapting and using eduscrum with real world projects. // In 2021 IEEE Frontiers in Education Conference (FIE). October, pp. 1-8.
- ⁹ Niemelä, M. (2020). Game learning analytics for understanding reading skills in transparent writing system. / M.Niemelä, T.Kärkkäinen, S.Äyrämö, [et all] // British Journal of Educational Technology. 51(6), p.2376-2390.
- ¹⁰ Pesovski, I., Vorkel, D., Trajkovik, V. (2024). AI-Powered Education: Rethinking the Way Programming is Taught Using AI Tools and Reversed Bloom's Taxonomy. // In 2024 21st International Conference on Information Technology Based Higher Education and Training (ITHET). November, pp. 1-6.
- ¹¹ West, M. (2023). An Ed-Tech Tragedy? Educational technologies and school closures in the time of COVID-19. / M.West. UNESCO Publishing.
- ¹² Zaripov, N. (2025). Mathematical and statistical analysis of students' mastery levels in the programming environment for teaching informatics and digital technologies. // AIP Conference Proceedings. 3268(1).