

Instrumental and executive system of training and testing

Aleksandr P. Polishchuk¹

¹Kryvyi Rih State Pedagogical University, 54 Gagarin Ave., Kryvyi Rih, 50086, Ukraine

Abstract. One of the ways to increase the efficiency of computer support of the educational process is to provide the subject teacher with a simple and easy-to-use interactive instrumental and executive system with a set of educational materials and test tasks for testing knowledge. The presence of two subsystems – the subsystem of learning and the subsystem of testing and performance accounting allows the teacher to solve the problem of simultaneous questioning in the classroom. The student, in turn, gets the opportunity to re-test independently to correct his unsatisfactory grades on certain topics already covered. Pedagogical experience shows that there were many such systems; most of them were focused on learning programming. Today, almost all training systems created in the 1960s, except PLATO (Programmed Logic for Automated Teaching Operations), have no practical significance. In their didactic capabilities, they differed little from the systems that used the simplest technical learning tools and provided a rigid, virtually excluding dialogue determination of student activity. However, it was the first developments that stimulated interest in computer learning, and the development of hardware and software for personal computers led to greater opportunities for their use in learning.

Keywords: educational process, instrumental and executive system of training and testing

Одним з напрямів підвищення ефективності комп'ютерної підтримки учбового процесу є надання викладачу-предметнику простої та зручної у користуванні інтерактивної інструментально-виконавчої системи з набором навчальних матеріалів та тестових завдань для перевірки знань. Наявність двох підсистем – підсистеми навчання та підсистеми тестування й обліку успішності дозволяє викладачеві вирішити проблему одночасового опитування при проведенні занять у комп'ютерному класі. Учень, в свою чергу, отримує можливість самостійного повторного тестування для виправлення не задовільняючих його оцінок з тих чи інших вже опрацьованих тем.

Педагогічний досвід свідчить, що таких систем було чимало; більшість з них були зорієнтовані на навчання програмуванню. Сьогодні майже всі створені у 60-і роки навчальні системи, крім PLATO (Programmed Logic for Automated Teaching Operations – програмована логіка для автоматизованих навчальних операцій [3]), не мають практичного значення. За своїми дидактичними можливостями вони мало відрізнялись від систем, що використовували найпростіші технічні засоби навчання і передбачали жорстку, практично виключаючу діалог детермінацію діяльності учнів. Проте саме перші розробки стимулювали інтерес до комп'ютерного навчання, а розвиток технічного і програмного забезпечення персональних комп'ютерів призвів до розширення

KMITO 1999: Conference on Computer Simulation and Information Technology in Education, April 19–21, 1999, Kryvyi Rih, Ukraine

✉ apol@cabletv.dp.ua (A. P. Polishchuk)



© Copyright for this paper by its authors, published by Academy of Cognitive and Natural Sciences (ACNS).

This is an Open Access article distributed under the terms of the Creative Commons License Attribution 4.0 International (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



ACNS Conference Series: Social Sciences and Humanities

можливостей їх використання у навчанні.

Не випадково в останні роки ми спостерігаємо стійкий інтерес до різного роду інструментальних оболонок типа Framework (на базі якої в Центральній-Міській гімназії м. Кривого Рога досить успішно вивчається курс комп'ютерного моделювання [7]), однак реалізація в них учбових задач відбувається не завжди адекватно, а найчастіше їх взагалі не можна втиснути у рамки цих систем. Інший їх різновид – авторські оболонки – як правило, призначені для якщо не разового використання, то для вузького кола користувачів. Більш того, навіть з такими системами познайомиться практично неможливо з тієї причини, що вони звичайно не виходять за межі учбового закладу, у якому розроблені.

Звичайною вимогою є те, що інструментальна частина системи повинна обслуговувати функції створення, знищення, коректування учбових матеріалів і тестових завдань, щоб кожен викладач мав змогу формувати виучуваний курс під свої методи і конкретні умови праці. У той же час поставка “порожніх” оболонок, навіть дуже гарних, малоефективна, оскільки процес їх наповнення вимагає витрат часу і вимагає замислюватись над тим, що саме ти хочеш зробити (наприклад, закладений нами у розроблену оболонку елементарний курс інформатики складається з 100 уроків й близько 1000 тестів загальним обсягом більше 1 Мб).

То ж як усунути чи пом'якшити зазначені вище недоліки? Насамперед, така система повинна бути гарно документованою і мати хоча б примітивну систему допомоги, бажано контекстно-залежну. Вона має бути відкритою і здатною до обміну даними з іншими програмами, не змушуючи користувача працювати з навчальним матеріалом лише в ній. І, звичайно ж, вона повинна мати інтуїтивно зрозумілі засоби керування, наприклад, в стандарті Common User Access фірми IBM [6].

Розроблена нами система TEST в одному з своїх варіантів є повноцінною Windows-аплікацією і в цьому варіанті може функціонувати на IBM-сумісних PC-386 та вище під керуванням Windows for Workgroups 3.11+. Це означає, що вона може виконуватися одночасово з іншими програмами, не заважаючи їм, а користувач може переходити за необхідності у будь-яку з них. Так, скажімо, при вивченні курсу чисельних методів типовими програмами, що сумісно працювали на одній ЕОМ, були

- Диспетчер програм (progman) – оболонка, з якої відбувається запуск усіх інших програм
- Pri/Sec (ruslat) – русифікатор Windows for Workgroups
- Borland C++ 3.1 for Windows (bcw) – інтегроване середовище мови C++, у якому виконувалися завдання з курсу чисельних методів
- Інструментально-виконавча система навчання та тестування (test), яка містила теорію з методу, завдання на лабораторну роботу, методичну допомогу, а в особливо важких випадках і окремі фрагменти програмної реалізації виучуваного методу [5] (figure 1).

Система пропонує користувачу 2 основних режими роботи: “Розробка” та “Експлуатація” (навчання та тестування). Режим розробки призначений для викладача і захищений від випадкового та несанкціонованого використання кнопкою пароллю, що

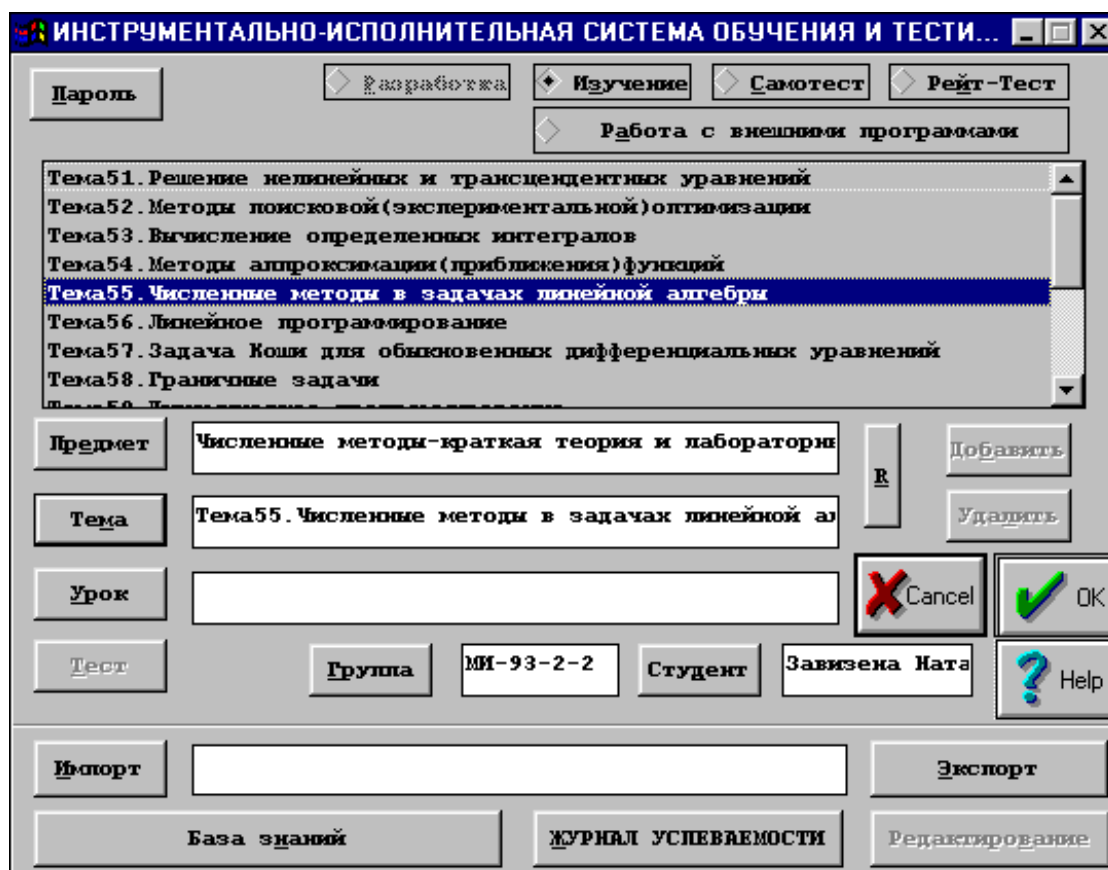


Figure 1: Інструментально-виконавча система навчання та тестування.

перевіряє наявність у дисководі ключового носія – дискети, а у разі її відсутності запитує пароль з клавіатури. Запуск системи здійснюється з DOS (з побіжним автоматичним запуском Windows) або безпосередньо з Windows. Windows система у користуванні попервах складна, тому запуск з Windows-програми з DOS була включена як можливість, що полегшує її експлуатацію. Як показав досвід, для успішного початку курсу лабораторних робіт з чисельних методів у Windows студентам, зовсім з нею не знайомим, досить однієї-двох пар. Після запуску встановлюється режим експлуатації. В системі відсутні багаторівневі меню вибору режимів та різних установок, такі звичні для складних систем – з метою спрощення всі органи керування зосереджені в головному діалоговому вікні у вигляді панелі керування з набором кнопок та текстових субвікон. Такого роду інженерний підхід – створення пульту або панелі керування – дозволяє постійно мати перед очима режим роботи, вибрану предметну область, групу та ім'я студента, що, як показав досвід користування, прискорює роботу, на відміну від меню-орієнтованих систем.

Для вибору виду робіт в режимі експлуатації у верхній частині панелі розташовано набір кнопок, що дозволяють обрати один з чотирьох видів діяльності, а у середній її

частині міститься велике вікно для відображення різних режимно-залежних списків (з можливістю вибору елемента списку) або журналу успішності для перегляду. Режим навчання реалізується після задання предметної області, вибір якої здійснюється за умовною трьохрівневою схемою ієрархії типа “Предмет – тема – урок”. На будь-якому рівні “натискання” кнопки “ОК” викликає перехід у вікно демонстрації навчального матеріалу одного або послідовно усіх уроків теми чи предмету (figure 2).

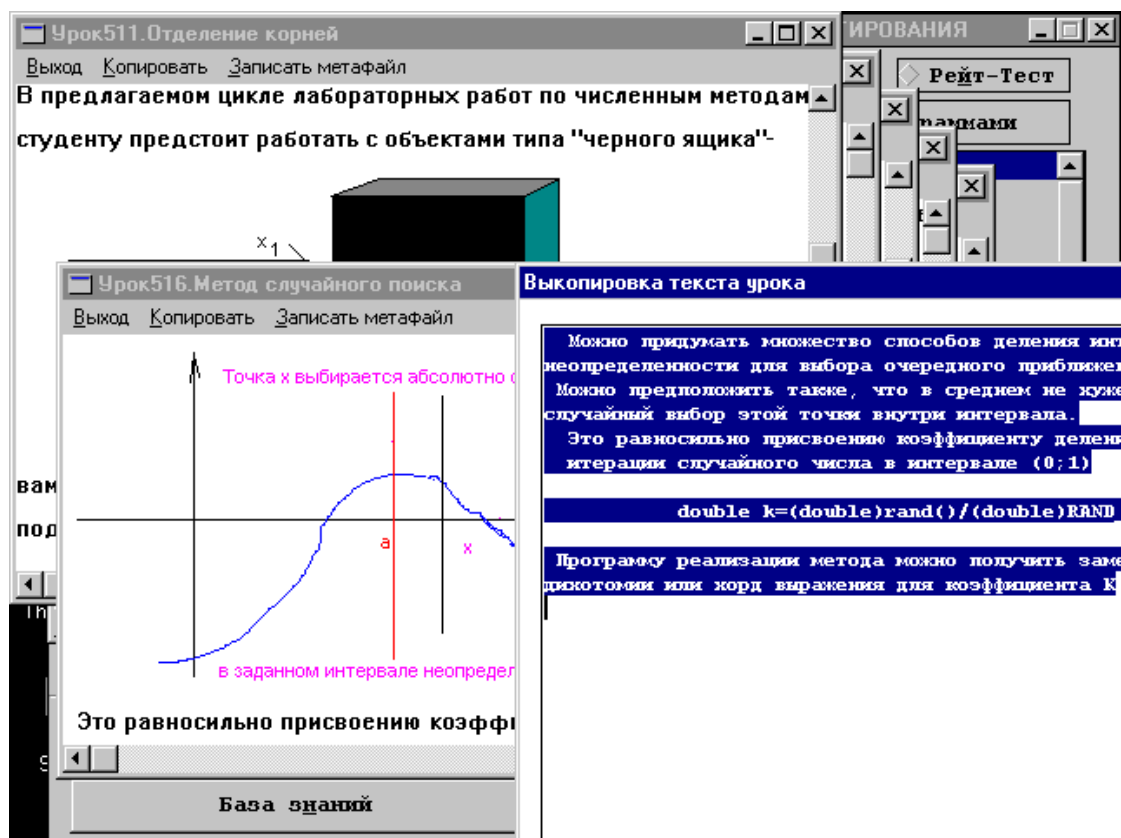


Figure 2: Відображення навчального матеріалу.

В режимі навчання доступні усі стандартні функції перегляду (редагування навчального матеріалу в цьому режимі заборонено), виділення та переносу блоків тексту через Clipboard – буфер обміну Windows, який є одним із засобів обміну даними між різними аплікаціями.

На жаль, оцінка, незважаючи на досвід Ш. О. Амонашвілі [1], досі залишається мірою знань, тому була здійснена спроба сумістити як тестування “для себе”, так й “для вчителя”. Режим “Самотестування” слугує допоміжним засобом навчання та анонімної перевірки знань без занесення оцінки до журналу успішності. В цьому режимі після вибору предметної області здійснюється перехід в діалогову панель тестування; набір текста відповіді здійснюється з клавіатури або копіюванням з буферу обміну (туди може бути занесена важка для запам’ятовування частина уроку або весь урок в режимі навчання,

так що навчальний матеріал при тестуванні, як кажуть, “під рукою”) (figure 3).

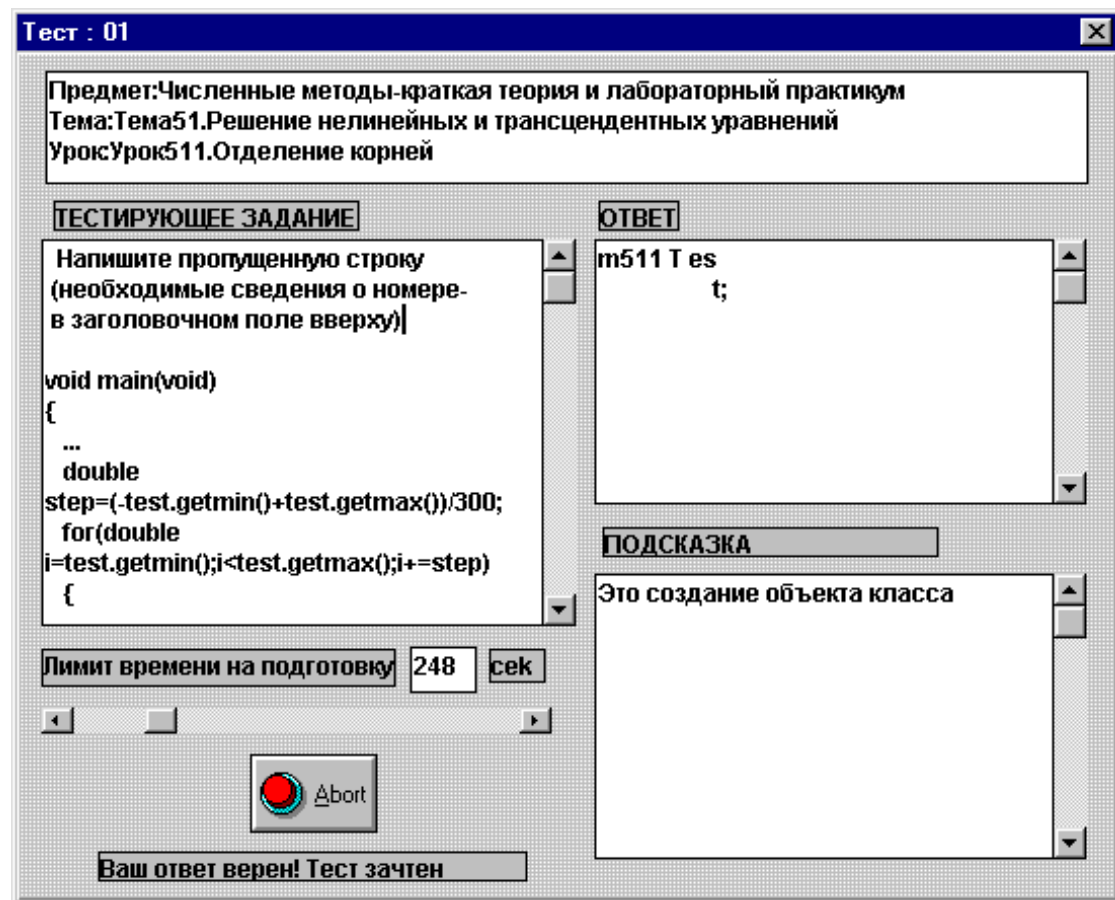


Figure 3: Режим тестування.

Система оцінювання тестового завдання м'яка та демократична – заощаджений час на підготовку до відповіді призводить до підвищення оцінки з заданим коефіцієнтом, а прострочений – до незаліку теста; за підказку ж доводиться розраховуватися деяким зниженням оцінки, яка виставляється після проходження обумовленої кількості тестів для даного уроку (не менше трьох).

Аналіз тексту відповіді – найбільш важка і відповідальна частина системи. Якщо б вона була закритою і обмежувалась однією областю знань, то проблем би не виникло. У літературі згадується, наприклад, система SOPHIE (sophisticated instructional environment). Це імітаційна програма, призначена для навчання пошуку несправностей у електронних ланцюгах [2]. В межах даної дуже вузької області ця програма дозволяє аналізувати відповідь, вказуючи не лише на помилку, а й на послідовність дій, що спричинили її. Сьогодні не існує ефективних методів оцінки смислового змісту відповіді за довільною тематикою, а тому доводиться покладатися на формальний аналіз тексту (збіг з еталоном). Існує ряд класифікацій тестів, але всі вони, від примітивних вибірових до конструкцій

з послідовності слів та фраз, набираються з клавіатури і первинним матеріалом для аналізу завжди є текст.

Очевидно, що випадкова помилка в одному символі при наборі не повинна призводити до незліку тесту. Скажімо, возьмемо який-небудь першоквітневий тест на тему "Саме...":

Питання : Напишіть прізвище російської радянської актриси, що складається з більш як 30 букв.

Еталонна відповідь : Архневолоточерепопіндрюковська

Найзаядліші балетомани навряд чи з першого разу зможуть написати його правильно (не кажучи вже про те, щоб вимовити). Крім того, в окремих випадках доводиться допускати можливість перестановки слів у фразі, тому в розглядуваній системі використовується багаторівневий аналіз. Якщо після видалення пропусків та переведень рядків немає повного співпадання з еталоном, для тестів із зведеним флагом складності виконується додатковий змістовний аналіз. Певна річ, у тестах з числовими відповідями, відповідями з визначенням номеру вірного (хибного) варіанту і таке інше зазначених проблем не виникає.

Режим рейтинг-тестування з занесенням оцінок до журналу реалізується після вибору групи та прізвища у списку, які за необхідності поповнює викладач. Різниця між цими режимами несуттєва, тому що студенту, який жодного разу не тестувався, за всі уроки у журналі стоїть оцінка "2", і його мета – не набирати гарні оцінки, щоб, протестувавшись на "відмінно" з одного уроку, одержати вищий бал з усього предмету (як це зроблено у деяких оболонках), а виправляти ці двійки. Такий підхід може здатися не дуже гуманним, однак альтернативи йому нема – тільки так можна забезпечити проходження учнем тестів якщо не за всіма, то хоча б з більшої частини уроків. Завдяки такому підходу перед ним не існує психологічного бар'єру, пов'язанного з можливістю отримання поганої оцінки – вона в нього вже стоїть.

Режим "Робота з імпортованими програмами" призначений для об'єднання "під одним дахом" усіх необхідних для вивчення курсу допоміжних інструментальних засобів. Це можуть бути навчальні програми з окремих розділів, анімаційні програми та навчаючі ігри, транслятори з мов програмування і взагалі будь-які програмні засоби.

Режим "Розробка" стає доступним після натиснення кнопки "Пароль" при наявності в дисководі ключової дискети або при її відсутності після набору пароля. Відмінність режиму розробки від режиму експлуатації полягає у доступності функцій коректування та редагування списків предметів, тем, уроків, тестов, груп та їх складу, імпортованих програм, текстів уроків і тестових завдань. Для формування уроків можуть використовуватися вже готові текстові файли в кодировці ANSI чи OEM (після перекодування утилітою fconvert або через стандартний Windows-редактор Write [Word-Pad] і буфер обміну). Ця можливість виявляється не тільки корисною, а й незамінною у тих випадках, коли курс лекцій вже набрано раніше. В оболонці також передбачена можливість експортування навчального матеріалу у файли в ANSI чи OEM кодуванні (тільки текст) чи у форматі Windows Metafile (текст + графіка). Отримання твердої копії з файлу залишається на розсуд користувача.

Система є об'єктно-орієнтованою; доступ до об'єктів класів "Урок" та "Тест" на диску прямий завдяки невеликим за обсягом асоціативним словникам, які одразу

завантажуються до оперативної пам'яті і безперервно коректуються у процесі роботи. Для прискорення дискових операцій запис кожного разу іде у кінець відповідного файлу, так що у процесі роботи на диску може виявитися декілька копій одного й того уроку чи тесту різної свіжості. Щоб запобігти такого марнотратства дискової пам'яті (а іноді це сотні кілобайт), наприкінці сеансу роботи з системою користувачу пропонується пересортувати дані з метою вилучення "мертвих" даних, на які немає посилань у словниках. Це опціональна операція, оскільки при великих об'ємах даних вона віднімає деякий час.

Зараз обговорювана оболонка наповнена уроками та тестами з чотирьох предметів: "Чисельні методи", "Елементарний (ввідний) курс інформатики", "Основи наукових досліджень", "Програмування в Turbo Vision". Апробація системи проводилась у чотирьох групах студентів 3-го курсу (поток МІ-93) й, окрім усунення ряду помилок і недоробок, показала такі результати.

Засвоєння органів керування досягається за 10-20 хвилин за умови хоча б мінімального досвіду спілкування з Windows; у разі його відсутності зайняття розпочинаються з підручника до операційного середовища Windows. При серйозному відношенні викладача до оцінок у системному журналі успішності останній грає ключову роль в активному використанні студентами навчальних матеріалів і відвідувань додаткових занять у комп'ютерному класі. Це особливо ефективно для студентів, що мають проблеми із засвоєнням курсу. Спостерігаються спроби багаторазового тестування, конспектування еталонних відповідей при незаліках тестів з метою майбутнього використання, активний пошук відповідей в текстах уроків; студент несамохіть втягується у процес "самонатаскування" з погано засвоєних розділів (у вільний від занять час). Інший ефект – підвищення продуктивності, наприклад, при навчанні програмуванню, виникає за рахунок копіювання фрагментів програм з наведених в уроках методичних матеріалів до інтегрованого середовища розробки, що дає можливість підвищити складність завдань з програмування шляхом "збирання" програм з рекомендованих фрагментів. Крім того, інструментально-виконавча система передбачає

1. Індивідуальний темп навчання, при якому студент може повернутися до будь-якої методичної рекомендації та прикладу, ще раз пройти погано засвоєний матеріал [4].
2. Диференційоване тестування, що дозволяє встановити рівень вимог студента щодо оцінки.
3. Незалежність від змісту відповіді при тестуванні та матеріалу при вивченні дає змогу наповнювати систему довільним матеріалом (як приклад можна навести курс "Основи наукових досліджень" та батарею тестів з педагогіки).

References

- [1] Amonashvili, S., 1989. Non-directive teaching and the humanization of the educational process. *Prospects*, 19(4), pp.581–590. Available from: <https://doi.org/10.1007/BF02206752>.
- [2] Brown, J.S., Burton, R.R. and Bell, A.G., 1975. SOPHIE: A step toward creating a reactive

- learning environment. *International Journal of Man-Machine Studies*, 7(5), pp.675–696. Available from: [https://doi.org/10.1016/S0020-7373\(75\)80026-5](https://doi.org/10.1016/S0020-7373(75)80026-5).
- [3] Control Data Corporation, 1981. PLATO User's Guide. Available from: http://www.bitsavers.org/pdf/cdc/plato/97405900C_PLATO_Users_Guide_Apr81.pdf.
- [4] Gergei, T. and Mashbits, E.I., 1986. Psychological and pedagogical problems of effective computer use in the educational process. *Soviet Education*, 28(10-11), pp.213–229. Available from: <https://doi.org/10.2753/RES1060-9393281011213>.
- [5] Polishchuk, A.P. and Semerikov, S.A., 2022. Concept of the course “Numerical methods in object methodology”. *ACNS Conference Series: Social Sciences and Humanities*, 1, p.01003. Available from: <https://doi.org/10.55056/cs-ssh/1/01003>.
- [6] Taylor, K., 1990. IBM Systems Application Architecture: Common User Access from first principles. *Computing & Control Engineering Journal*, 1, pp.123–127(4). Available from: https://digital-library.theiet.org/content/journals/10.1049/cce_19900033.
- [7] Teplytskyi, I.O., 2022. Physics models in the course “The basics of computer simulation”. *ACNS Conference Series: Social Sciences and Humanities*, 1, p.01002. Available from: <https://doi.org/10.55056/cs-ssh/1/01002>.