

**ВНЗ Укоопспілки «Полтавський університет економіки
і торгівлі»**

*Кафедра математичного моделювання та соціальної
інформатики*



*Присвячується 60-й річниці
освітньої діяльності університету*

**КОМП'ЮТЕРНІ НАУКИ І ПРИКЛАДНА
МАТЕМАТИКА
(КНіПМ-2021)**

**МАТЕРІАЛИ НАУКОВО-ПРАКТИЧНОГО
СЕМІНАРУ
Випуск 6**

січень-червень 2021 р.

Полтава
2021

КОМП'ЮТЕРНІ НАУКИ І ПРИКЛАДНА МАТЕМАТИКА (КНіПМ-2021): матеріали науково-практичного семінару. Випуск 6 / за ред. Ємця О.О. – Полтава: Кафедра ММСІ ПУЕТ, 2021. – 106 с.

Збірник матеріалів науково-практичного семінару містить сучасну проблематику в таких галузях інформатики та системних наук, як теоретичні основи інформатики та кібернетики, математичне моделювання та обчислювальні методи, математичне та програмне забезпечення обчислювальних машин і систем, системний аналіз і теорія оптимальних рішень. Представлено тези доповідей, що відображають проблеми сучасної підготовки фахівців з комп'ютерних наук, прикладної математики, системного аналізу та комп'ютерних інформаційних технологій.

Збірник присвячується 60-й річниці освітньої діяльності університету.

*Матеріали друкуються в авторській редакції мовами оригіналів.
За виклад, зміст і достовірність матеріалів відповідають автори*

Ум. друк. арк. 6,62
©Кафедра ММСІ ПУЕТ, 2021

ЗМІСТ

Пилипченко В. С., Ємець О. О. Про тренажер, що навчає створенню моделі та розв'язуванню в «Пошуку рішень» на прикладі задачі «Максимізація щомісячного прибутку підприємства».....	5
Цехмейстер В. О., Ємець О. О. Максимізація прибутку кондитерської фабрики: алгоритмізація та програмування елементів тренажеру з побудови математичної моделі.....	9
Деркач Р. А., Ємець О. О. Ігрові задачі комбінаторного типу: програмне забезпечення і дослідження.....	13
Стельнік А. І., Черненко О. О. Розробка програмного забезпечення тренажеру з теми «регулярні мови» дистанційного навчального курсу «Теорія програмування».....	16
Фесенко Д. І., Олексійчук Ю. Ф. Програмний комплекс для виконання та оцінювання завдань з теми «Аналіз алгоритму сортування злиттям» дисципліни «Аналіз алгоритмів».....	21
Клименко М. М., Чілікіна Т. В. Створення програмного забезпечення з теми «Логічні оператори $\rho h\rho$ ».....	25
Купченко О. В., Парфьонова Т. О. Створення тренажеру з теми «Коди із виправленням помилок».....	35
Олефіренко В. В. Розробка елементів тренажеру з теми «Сортування включеннями з лінійним та бінарним пошуком» дистанційного навчального курсу «Алгоритми та структури даних» та розробка його програмного забезпечення.....	37
Бибка Б. М., Ємець Ол-ра О. Тренажер «Побудова блок-схем алгоритмів циклічної структури на прикладі циклу While».....	40
Бурко А. О. Створення тренажеру дистанційного навчального курсу «Дискретна математика» з теми «Алгебра Жегалкіна, способи побудови поліномів Жегалкіна».....	51
Дудник Д. А., Парфьонова Т. О. Розробка програмного забезпечення для тренажера з теми «Теорема додавання та множення ймовірностей випадкових подій» дистанційного навчального курсу «Теорія ймовірностей і математична статистика».....	54
Закіров Р. Р. Програмне забезпечення тренажеру з теми «Лямбда-вирахування і типізація» дистанційного навчального курсу «Теорія програмування».....	60

Іжевський Д.О. Програмування елементів тренажера «Граматика. Мови, що задаються граматиками» дистанційного навчального курсу «Теорія програмування».....	63
Пильник С. О. Створення програмного забезпечення з теми «Арифметичні оператори Assembler».....	66
Рак А. О. Створення елементів тренажера дистанційного навчального курсу «Математичний аналіз» з теми «Обчислення границь функції однієї змінної».....	69
Рижков А. В., Ємець Є.М., Ємець О.О. Тренажер «Побудова блок-схем алгоритмів циклічної структури на прикладі циклу repeat ... until».....	72
Лукіна В.В., Черненко О.О. Розробка програмного забезпечення тренажеру з теми «Магазинні автомати» дистанційного навчального курсу «Теорія програмування».....	85
Ткаленко Ю.О. Розв’язування диференціальних рівнянь першого порядку.....	88
Малишева Д. В. Цикл for в Excel VBA.....	90
Борута І. В., Парфьонова Т.О. Тренажер «Відношення. Область визначення, область значень, граф, матриця відповідності, переріз за елементами» дистанційного навчального курсу «Дискретна математика».....	91
Lartey W. H., Oleksiichuk Yu. F. Algorithmization and development of simulator on the topic “Cycles in java” of the discipline “Programming”.....	95
Chagonda N. S. Development of a simulator of the distance learning course.....	98
Darboe A. Elements for the simulator on the topic "Algorithm for onducting the left factorization of grammar" of the distance learning course "Programming theory".....	100
Muhammad A. Software elements for the simulator on the topic "Predictive parsing: scheme, principle of operation, application" of the distance learning course "Programming theory".....	103
Алфавітний покажчик авторів.....	106

УДК 004+519.852

**ПРО ТРЕНАЖЕР, ЩО НАВЧАЄ СТВОРЕННЮ МОДЕЛІ
ТА РОЗВ'ЯЗУВАННЮ В «ПОШУКУ РІШЕНЬ» НА
ПРИКЛАДІ ЗАДАЧІ «МАКСИМІЗАЦІЯ ЩОМІСЯЧНОГО
ПРИБУТКУ ПІДПРИЄМСТВА»**

В. С. Пилипченко, студент гр. КН 6-41, спеціальності
«Комп'ютерні науки» ВНЗ Укоопспілки ПУЕТ
vitaliy230477@gmail.com

О. О. Ємець, д.ф.-м.н., професор
ВНЗ Укоопспілки ПУЕТ
yemetsli@ukr.net

В доповіді розглядаються питання створення тренажеру по навчанню моделюванню, а також розв'язуванню оптимізаційної задачі в надбудові Excel «Пошуку рішень».

Pylypchenko V. S., Iemets O.O. About the simulator that teaches creation of a model and solution in "Solver" on the example of the problem "Maximization of the monthly profit of the enterprise".

The issues of creating a simulator for modeling training, as well as solving the optimization problem in Excel "Solver" are considered in the report.

Ключові слова: ОПТИМІЗАЦІЯ, ТРЕНАЖЕР, ПОШУК РІШЕНЬ, МОДЕЛЬ.

Keywords: OPTIMIZATION, SIMULATOR, SOLVER, MODEL.

Актуальність створення тренажера обумовлена тим, що його використання можливе в будь якому місці, а також при наявності комп'ютера та первинних даних (програми тренажеру, MS Excel).

Об'єкт розробки – створення програмного забезпечення дистанційного навчання.

Предмет розробки – програма тренажер для навчання складання математичної моделі на прикладі планування роботи фірми.

Мета роботи – розробка алгоритму та програми тренажеру з теми «Максимізація щомісячного прибутку підприємства: програмна реалізація тренажера дистанційного курсу «Проектне навчання з курсу «Методи оптимізації та дослідження операцій»»

В тезах викладено деякі кроки алгоритму тренажеру побудови математичної моделі. В тренажері використовуються питання з кількома варіантами відповідей. Серед них лише один правильний. Якщо відповідь не правильна, то студент отримує інформативне вікно про неправильність відповіді. Після закриття цього вікна, і зробивши висновки, студент вибирає інший варіант відповіді. У тому випадку, коли відповідь правильна, відбувається перехід на наступне питання.

На нульовому кроці виводиться постановка задачі, на якій студент вчиться складати модель.

Крок 0. На початку роботи, студент отримує умову задачі. Підприємство має два види устаткування A_1, A_2 , на якому можна виробляти продукцію 4 видів, збут якої необмежений, тому підприємство само планує асортимент і величину випуску продукції. Сировина також може постачатися в необмеженій кількості. В таблиці 1 наведений місячний фонд часу використання кожного виду устаткування, витрати часу на одиницю виробу і величину прибутку від реалізації одиниці продукції кожного виду. Треба знайти план випуску продукції, щоб місячний прибуток був максимальний.

Таблиця.1.

	B_1	B_2	B_3	B_4	Місячний фонд використаного часу
A_1	1	0	3	6	120
A_2	2	3	2	0	230

Прибуток за одиницю виробу	45	50	20	60	
---	----	----	----	----	--

За цими даними студент повинен провести роботу з тренажером.

Наступний крок ілюструє вибір однієї відповіді з декількох.

Крок 5. Які ресурси обмежують виробництво продукції?

- а) тільки сировина;
- б) тільки час;
- в) тільки сировина і час;

Правильна відповідь: б).

Наступний крок ілюструє реалізацію вводу інформації студентом.

Крок16. На аркуші Excel в колонках A, B, C, D, E, F, в рядках 2-3 вставляємо числа з умови задачі. З правого боку таблиці в колонці G, записуємо формули для лівої частини обмежень місячного фонду часу. В активному полі студенту дається можливість записати першу формулу. Програма перевіряє її правильність.

Правильна відповідь: $B2*B8+C2*C8+D2*D8+E2*E8$

DELTA	X	✓	f_x	$=B2*B8+C2*C8+D2*D8+E2*E8$			
	A	B	C	D	E	F	G
1		B1	B2	B3	B4	Місячний фонд часу	Використаний час
2	A1	1	0	3	6	120	$=B2*B8+C2*C8+D2*D8+E2*E8$
3	A2	2	3	2	0	230	0
4	Приб.з одиниці		45	50	20	60	
5							
6							
7		x1	x2	x3	x4	міс.фонд часу	
8		0	0	0	0	0	
9	f=	0					

Рис 1.

В доповіді викладено деякі кроки алгоритму розробленого тренажеру побудови математичної моделі та розв'язування задачі в пошуку рішень Excel.

Література

1. Ємець О. О. Методи оптимізації та дослідження операцій: навчальний посібник / О. О. Ємець. - Полтава : ПУЕТ, 2019. – Ч.1. – 245 с. – Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/8598>
2. Ємець О. О. Про розробку тренажерів для дистанційних курсів кафедрою ММСІ ПУЕТ / О. О. Ємець // Інформатика та системні науки (ІСН-2015): матеріали VI Всеукр. наук.-практ. конф. за міжн. участю (м. Полтава, 19-21 березня 2015 р.) / за ред. Ємця О. О. – Полтава: ПУЕТ, 2015. – С.152-161. – Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/2488>
3. Kononets, N. Future teachers resource-based learning system: Experience of higher education institutions in Poltava city, Ukraine /N. Kononets, O. Ilchenko, V. Mokliak // Turkish Online Journal of Distance Education.– 2020.– V.21,is.3.– P. 199–220. – Режим доступу: <https://dergipark.org.tr/en/download/article-file/1181911>

**МАКСИМІЗАЦІЯ ПРИБУТКУ КОНДИТЕРСЬКОЇ
ФАБРИКИ: АЛГОРИТМІЗАЦІЯ ТА ПРОГРАМУВАННЯ
ЕЛЕМЕНТІВ ТРЕНАЖЕРУ З ПОБУДОВИ
МАТЕМАТИЧНОЇ МОДЕЛІ**

В. О. Цехмейстер, студент гр. Кн-м 11, спеціальності
«Комп'ютерні науки»
volavu02@gmail.com

О. О. Ємець, д.ф.-м.н., проф.
Вищий навчальний заклад Укоопспілки «Полтавський універ-
ситет економіки і торгівлі»
yemetsli@ukr.net

Запропоновано тренажер з теми: «Максимізація прибутку кондитерської фабрики: алгоритмізація та програмування елементів тренажера з побудови математичної моделі»

Tsekhmeister V.O., Iemets O.O. Simulator for the topic "Maximizing the profit of a confectionery factory: algorithmization and programming of simulator elements for building mathematical models. Coaching programs are written for this algorithm

Ключові слова: ТРЕНАЖЕР, АЛГОРИТМІЗАЦІЯ, ПРИБУТОК, МАТЕМАТИЧНА МОДЕЛЬ.

Keywords: SIMULATOR, ALGORITHMIZATION, MATHEMATICAL MODEL

В тезах викладена постановка задачі, короткий опис тренажера і частина алгоритму роботи тренажера. Метою роботи є розробка алгоритму та програмного забезпечення тренажера з теми «Максимізація прибутку кондитерської фабрики: алгоритмізація та програмування елементів тренажера з побудови математичної моделі».

При аналізі та реалізації даної теми використано [1], також проаналізовано подібні навчальні тренажери [2] та опрацьовано матеріал підручника [3].

Тренажер матиме у собі сторінку із теоретичними відомостями та із практичним симулятором, у якому потрібно буде приймати рішення для збільшення прибутку кондитерської фабрики.

Теоретична частина міститиме економічні та аналітичні матеріали, що допоможуть у проходженні практично частини тренажеру. Практична частина міститиме тести, які у собі матимуть певні економічні та підприємницькі проблеми, із якими може зіткнутись підприємство у своїй діяльності.

В результаті тренажер на основі ваших результатів спрогнозує ваш максимальний прибуток. Таким чином, можна сказати, що це є програмний симулятор підприємця, який дає змогу користувачу перевірити свої сили та вирішити економічні, соціальні та підприємницькі проблеми різного типу.

Алгоритм роботи тренажеру описано далі.

Наразі представлено відрізок із цілого модуля завдань та проблем, які є перед користувачем, та на основі яких рахуватиметься прогнозований прибуток у відсотковій величині.

На початку роботи відсоток вашого прибутку від роботи фабрики складатиме 0 відсотків та лише від дій користувача він може стати максимальним.

На початку роботи тренажера виводиться умова:

Крок 0. Ви є новим директором кондитерської фабрики, ваше завдання вирішуючи різного роду проблем досягти максимального доступного прибутку кондитерської фабрики.

Крок 1. Виводиться умова: «Фінансова ситуація кондитерської фабрики у найгіршому стані за увесь період її роботи, яким буде ваш перший крок, що дасть змогу поліпшити ситуацію» При виборі варіанту вам потрібно задовільнити такі умови:

1. Щоб це не вплинуло на робочий процес.
2. Не сплинуло на працівників.
3. Не вплинуло на якість продукції.

Варіанти відповіді:

- А) Розширювати ринки збуту.
- Б) Знизити заробітню плату працівникам.
- В) Здійснювати закупівлю більш дешевої сировини

Кожна відповідь має свою кількість балів, які будуть перетворені у відсотки.

Якщо обрати варіант А ви отримаєте – 100 балів, варіант Б) –10балів, варіант В) –100 балів

Крок 2. Виводиться умова: Вам вдалося подолати фінансову кризу шляхом розширення ринків продажу, але ваш товар не впізнають, та він не є популярним. Ваші дії:

- А) Розмістити рекламу у ЗМІ та інтернеті.
- Б) Розробити акційні пропозиції та знижкову систему
- В) Роздавати листівки самостійно.

Кожна відповідь має свою кількість балів, які будуть перетворені у відсотки.

Якщо обрати варіант А , ви отримаєте 60 балів, варіант Б) – 30 балів, варіант В) – 10 балів

При видачі балів братимуться до уваги, як фінансові так і моральні аспекти діяльності.

У загальній кількості буде близько 10-20 ситуацій різного направлення.

Після проходження тренажеру, буде згенеровано прибуток у відсотках та оцінка.

Максимальним вважатиметься відсоткова міра прибутку 89-99 відсотків.

Література

1. В.В Вітлінський, Т.О Терещенко, С.С. Савіна Економіко-математичні методи і моделі оптимізація. — М. : Видавництво Київ «КНЕУ», 2016. — 303 с – Режим доступу: <https://scholar.google.com.ua/citations?user=L-4Y0ukAAAAJ&hl=ru>
- 2.Ємець О. О. Інформатика та системні науки (ІСН-2015) [Електронний ресурс] : матеріали VII Всеукр. наук.-прак. конф. за міжнар. участю, (м. Полтава, 19–21 берез. 2015 р.) / О.

- О. Ємець. – Полтава : ПУЕТ, 2015. – 402 с. – Режим доступу:
http://math.accent.kiev.ua/book/01/png_hm/01_3_18_png.htm
3. Л.В. Забуранна, Н.В Попрозман, Н.А. Клименко Оптимізація
методи та моделі: Видавництво Київ «КНЕУ», 2014. — 372 с. –
Режим доступу:
https://nubip.edu.ua/sites/default/files/u104/Підручник_18.pdf. – 29
березня 2021р.

**ІГРОВІ ЗАДАЧІ КОМБІНАТОРНОГО
ТИПУ: ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ І ДОСЛІДЖЕННЯ**

***Р. А. Деркач**, студент гр. КН б-41, спеціальності
«Комп'ютерні науки»*

*Вищий навчальний заклад Укоопспілки «Полтавський
університет економіки і торгівлі»
romanderkach1998@gmail.com*

***О.О.Ємець**, д.ф.-м.н., професор
ВНЗ Укоопспілки ПУЕТ
yemetsli@ukr.net*

*Запропоновано програмну реалізацію ітераційного методу типу
Брауна-Робінсон для ігрових задач комбінаторного типу з
обмеженнями-перестановками для обох гравців.*

*Derkach R.A, Iemets O.O. Game problems of combinatorial type:
software and research*

*A software implementation iterative method type Brown-Robinson
for game problems of combinatorial type of limited-permutations for
both players*

Ключові слова: ІГРОВІ ЗАДАЧІ, ЦІНА ГРИ,
КОМБІНАТОРНІ ЗАДАЧІ.

Keywords: GAME PROBLEMS, PRICE OF THE GAME,
COMBINATORIAL PROBLEMS.

Актуальність теми пояснюється тим, що серед задач комбінаторної оптимізації [1] все популярнішими стають задачі розв'язування конфліктних ситуацій, котрі відносяться до класу задач комбінаторної оптимізації ігрового типу з обмеженнями на стратегії гравців, що визначаються комбінаторними

конфігураціями, в тому числі множинами перестановок та розміщень (див., зокрема [2-4]). Для подібних задач є необхідними дослідження уже існуючих методів розв'язування, можливостей їхньої модифікації та проведення різноманітних експериментів з метою визначення їх практичної ефективності.

Об'єкт розробки – програмне забезпечення для методу Брауна-Робінсон. Предмет розробки – консольні додатки, котрі реалізують ітераційний метод типу Брауна-Робінсон.

В роботі реалізована основна мета – створити програми, які реалізують ітераційний метод типу Брауна-Робінсон для розв'язування ігрових задач з обмеженнями-перестановками у обох гравців.

В [2] була побудована математична модель ігрової задачі з комбінаторними обмеженнями на стратегії двох гравців, що визначаються перестановками. Для знаходження оптимальної стратегії гравців, коли стратегії є комбінаторними конфігураціями, використано ітераційний метод типу Брауна Робінсон (див., зокрема [3,4]).

Були створені консольні додатки, котрі реалізують ітераційний метод типу Брауна-Робінсон. Також була здійснено порівняння результатів роботи методу в цих консольних додатках. Більш швидкою виявилась програма, реалізована на мові програмування C++, чим програма, реалізована на мові програмування Pascal.

Література

1. Стоян Ю.Г. Теорія і методи евклідової комбінаторної оптимізації / Ю.Г. Стоян, О.О. Ємець – К.: Ін-т системн. досліджень освіти, 1993. – 188 с.
2. Емец О.А. Исследование математических моделей и методов решения задач на перестановках игрового типа / О.А. Емец, Н.Ю. Устьян // Кибернетика и сист. анализ. – 2007.– №6 – С.103-114.
3. Iemets O.A. Iterative Method for Solving Combinatorial Optimization Problems of the Game-type on Arrangements / Oleg A. Iemets, Elena V. Olkhovskaja // Journal of Automation and Information Sciences. – 2011. – Vol. 43 , № 5 – P. 52–63.

- 4.Ємець О.О. Збіжність ітераційного методу розв'язування задач комбінаторної оптимізації ігрового типу з обмеженнями-переставленнями у обох гравців / О. О. Ємець, О. В. Ольховська // XVIII Міжнародна конф. з автоматичного управління «Автоматика-2011»: Матер. конф. 28 – 30 вересня 2011 р., Львів. – Львів: Вид-во Львів. політех., 2011. – С. 71-72.

**РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ
ТРЕНАЖЕРУ З ТЕМИ «РЕГУЛЯРНІ МОВИ»
ДИСТАНЦІЙНОГО НАВЧАЛЬНОГО КУРСУ «ТЕОРІЯ
ПРОГРАМУВАННЯ»**

*Стельнік А.І., магістр спеціальності «Комп'ютерні науки»
Черненко О.О., к.ф.-м.н., доцент – науковий керівник*

Ключові слова: тренажер, регулярні мови, теорія програмування.

Постановка проблеми. У наш час розвиток дистанційної освіти (ДО) є нагальною потребою для швидкого входження України до інформаційного суспільства, інтегрування її у європейську і світову спільноту. В Законі України про вищу освіту вже визначено місце ДО в системі національної освіти як рівноцінної форми навчання поряд із очною, заочною та екстернатом.

Для дистанційного курсу «Теорія програмування» потрібно розробити тренажер.

На початковому вікні повинна виводитися наступна інформація про тренажер:

- тема;
- прізвище, ініціали студента;
- академічна група студента;
- прізвище, ініціали керівника.

В тренажері згідно теми мають бути висвітлені такі питання:

1. Мови
2. Регулярні операції.
3. Граматики
4. Регулярні вирази.

Перед проходженням тренажеру слід користувачу дати можливість переглянути теоретичний матеріал по наведеним питанням. При невірній відповіді на практичні завдання повинно виводитися спочатку попередження про помилку, а вже потім пояснення цієї помилки.

Так як програма може використовуватися при навчанні іноземних студентів пропонується реалізувати можливість зміни мови на початковому вікні.

Аналіз основних досліджень і публікацій. Відразу необхідно зазначити, що не існує двох однакових програм для дистанційного навчання. Можуть бути подібні методи та форми, проте, в цілому навчання завжди буде відрізнятися. Це властиво будь-якій загальноосвітній чи вищій школі, де навіть при наявності єдиних стандартів та загальних вимог, фактичне навчання трохи відрізняється. Ці відмінності випливають з природних потреб навчальних закладів, де фактично працюють різні викладачі, діють різні академічні школи, встановлюється різний акцент на напрямку навчання [1].

Ефективний контроль знань в системі дистанційного навчання забезпечується поряд з іншими формами проведення підсумкових заходів сеансами тестування з використанням завдань різних типів.

Тип 1. Вибір однієї правильної відповіді. При відповіді на питання необхідно обрати лише один вірний варіант із запропонованих.

Тип 2. Вибір кількох правильних відповідей. При відповіді на питання необхідно обрати кілька вірних варіантів із запропонованих.

Тип 3. Встановлення відповідностей/підстановки. Тестове завдання на співставлення об'єктів та їх означень.

Тип 4. Встановлення правильної послідовності (порядок об'єктів). Тестове завдання, у якому необхідно встановити правильну послідовність елементів, дій, подій, операцій, слів у реченні тощо.

Тип 5. Заповнення пропусків («чіткі» або «нечіткі» підстановки). Тестове завдання без указаних можливих варіантів відповідей. Необхідно самостійно ввести в спеціальне поле відповідь (текстові або числові дані). [2].

Формулювання мети. Метою статті є ознайомлення з процесом розробки тренажеру з теми «Регулярні мови» дистанційного навчального курсу «Теорія програмування».

Виклад основного матеріалу дослідження.

Дистанційне навчання – це технологія, що базується на принципах відкритого навчання, широко використовує комп’ютерні навчальні програми різного призначення та створює за допомогою сучасних телекомунікацій інформаційне освітнє середовище для постачання навчального матеріалу та спілкування.

Дистанційне навчання має низку переваг перед іншими формами навчання. Так, практично не виходячи з дому чи не покидаючи свого робочого місця, можна підтримувати регулярний контакт з викладачем за допомогою телекомунікаційних технологій, у тому числі відеозв’язку, та одержувати структурований навчальний матеріал, представлений в електронному вигляді. Незначна за часом та обсягом частина навчального процесу дистанційної освіти може здійснюватися за очною формою (складання іспитів, практичні, лабораторні роботи тощо).

Метою виконаного проекту є розроблений алгоритм роботи тренажеру з теми «Регулярні мови» дистанційного навчального курсу «Теорія програмування».

В алгоритмі згідно теми висвітлено такі питання:

1. Мови
2. Регулярні операції.
3. Граматики
4. Регулярні вирази.

Після кожного блоку питань виводиться результат і пропонується перейти до наступного або повернутися на початкове вікно.

Кожне завдання може представляти собою відповідь на тестове питання або розв’язок прикладу. Наприклад, тестові питання:

Крок 1. Користувачу виводиться завдання і варіанти відповіді: «Множина всіх слів у алфавіті $\{a\}$ позначається».

- $\{a\}^* = \{\varepsilon, a, aa, aaa, \dots\} = \{a^n \mid n \geq 0\}.$
- $\{a\}^* = \{a, aa, aaa, \dots\} = \{a^n \mid n \geq 0\}.$

$$\bullet \quad \{a\}^* = \{\varepsilon, a, aa, aaa, \dots\} = \{a^n \mid n \geq 0\}.$$

Якщо відповідь – перший варіант, то перехід на наступний крок. Якщо ні – перший раз виводиться попередження про помилку, потім пояснення.

Крок 2. Користувачу виводиться завдання і варіанти відповіді: « $\{a^n \mid n \text{ – непарне}\}$ позначає».

- множину непарної довжини в алфавіті $\{a\}$.
- мову слів непарної довжини в алфавіті $\{a\}$.
- обидві відповіді вірні.

Якщо відповідь – третій варіант, то перехід на наступний крок. Якщо ні – перший раз виводиться попередження про помилку, потім пояснення.

Розв'язок прикладів:

Приклад 1. Ідентифікатор є послідовністю букв і цифр, що починається буквою. Множина всіх ідентифікаторів у алфавіті $X = \{a, b, 1\}$ нескінченна. Якщо записати їх за зростанням довжини, то початок буде таким: $\{a, b, a1, aa, ab, b1, ba, bb, \dots\}$ [3].

Приклад 2. Мова $\{x^n y^n \mid n \geq 0\}$ описується граматикою

$$G_1 = (\{x, y\}, \{S\}, P, S).$$

Тут $P = \{S \rightarrow xSy, S \rightarrow xy\}$ [3].

Приклад 3. Граматикою для мови $\{x^m y^n \mid m, n \geq 0\}$ є $G_2 = (\{x, y\}, \{S, B\}, P, S)$.

Тут набір продукцій P має вигляд

$$\begin{array}{ll} S \rightarrow xS, & S \rightarrow y, \\ S \rightarrow yB, & B \rightarrow yB, \\ S \rightarrow x, & B \rightarrow y. \end{array}$$

Оскільки порожній рядок також належить мові, у набір P також входить продукція $S \rightarrow \varepsilon$.

Рядок $xxyuu$ генерується в такий спосіб:

$$S \Rightarrow xS \Rightarrow xxS \Rightarrow xxyB \Rightarrow xxyyB \Rightarrow xxyuu \text{ [3].}$$

Приклад 4. Множину продукцій грамматики

$$G_1 = (\{a, 1, 2\}, \{A, B, C, D\},$$
$$\{A \rightarrow BC, A \rightarrow BD, A \rightarrow B, B \rightarrow a, C \rightarrow 1, D \rightarrow 2\}, A)$$

можна переписати у вигляді

$$\{A \rightarrow B[C \mid D], B \rightarrow a, C \rightarrow 1, D \rightarrow 2\} \text{ [3].}$$

Висновки. В статті викладено результати огляду матеріалу з теми «Регулярні мови» та його подальшого використання при створенні тренажеру. Тренажер може використовуватися для навчання студентів в дистанційному курсі «Теорія програмування».

Література

1. Дистанційне навчання. Міфи і реальність / Міська спеціалізована молодіжна бібліотека «Молода гвардія» (МСМБ «Молода гвардія») [Електронний ресурс]. – Режим доступу: <http://msmb.org.ua/stily-zhittya/karyera-ta-osvita/distsiynne-navchannya-mifi-i-realynisty/>
2. Шовкоплас О. А. Звіт про науково-дослідну роботу «Моделювання навчального процесу вивчення економіко-математичних дисциплін з використанням комп'ютерних технологій» / О. А. Шовкоплас, О. О. Базиль, О. А. Літвіненко, О. С. Мазманішвілі – Суми: Вид-во СумДУ, 2015. – 53 с.
3. Черненко О.О. Електронний навчально-методичний посібник для самостійного вивчення навчальної дисципліни «Теорія програмування» для студентів напряму 6.040302 «Інформатика».

УДК 004.588

**ПРОГРАМНИЙ КОМПЛЕКС ДЛЯ ВИКОНАННЯ ТА
ОЦІНЮВАННЯ ЗАВДАНЬ З ТЕМИ «АНАЛІЗ
АЛГОРИТМУ СОРТУВАННЯ ЗЛИТТЯМ» ДИСЦИПЛІНИ
«АНАЛІЗ АЛГОРИТМІВ»**

*Д. І. Фесенко, студент гр. КН м-41, спеціальності
«Комп'ютерні науки та інформаційні технології»*

*Вищий навчальний заклад Укоопспілки «Полтавський
університет економіки і торгівлі»*

1804990@gmail.com

*Ю. Ф. Олексійчук, доцент кафедри математичного
моделювання та соціальної інформатики, к.ф.-м.н.*

*Вищий навчальний заклад Укоопспілки «Полтавський
університет економіки і торгівлі»*

olexijchuk@gmail.com

*В публікації розглядається програмний комплекс для
виконання та оцінювання завдань з теми «Аналіз алгоритму
сортування злиттям» дисципліни «Аналіз алгоритмів»*

*Fesenko D.I., Oleksiychuk Yu.F. Software package for performing
and evaluating tasks for the topic "Analysis of the merge sorting
algorithm" of the discipline "Analysis of algorithms". The
publication a software package for performing and evaluating tasks
are considered for the topic "Analysis of the merge sorting
algorithm" of the discipline "Analysis of algorithms".*

*Ключові слова: АЛГОРИТМ СОРТУВАННЯ ЗЛИТТЯМ,
АНАЛІЗ АЛГОРИТМІВ, ТРЕНАЖЕР*

*Keywords: MERGE SORTING ALGORITHM, ALGORITHM
ANALYSIS, SIMULATOR*

*В доповіді розглядається постановка задачі та програмний
комплекс для виконання та оцінювання завдань з теми «Аналіз
алгоритму сортування злиттям» дисципліни «Аналіз
алгоритмів»*

За останні роки було розроблено тренажери з різних тем дисципліни «Аналіз алгоритмів» [1-3] та інших дисциплін кафедри математичного моделювання та соціальної інформатики [4-6].

В рамках бакалаврської роботи було розроблено навчальний тренажер з теми «Аналіз алгоритму сортування злиттям» дисципліни «Аналіз алгоритмів» [7]. Задачею тренажеру є допомогти студенту детально розібратися, як працює алгоритм сортування злиттям [8].

Тренінг розпочинається з того, що з'являється екран, на якому міститься випадковий приклад, алгоритм злиття та поля для введення відповіді. При введенні відповіді в поля, з'являється вікно, яке інформує користувача про правильну або не правильну відповідь.

Головною задачею тренажеру є допомога студенту при самостійному вивченні відповідної теми. Але крім самонавчання актуальною також є проблема перевірки знань студентів різних форм навчання. Особливо це актуально в період карантину, коли неможливо стаціонарно навчатися та викладач не може швидко і якісно перевірити знання студента. До того ж оцінювання буде спонукати студента більш активно працювати із програмою. Тому є необхідність створення програмного комплексу з теми «Аналіз алгоритму сортування злиттям» для перевірки та оцінювання знань студента.

Програмний комплекс буде складатися з двох програм, а саме з програми для перевірки знань та програма для верифікації. Доступ до останньої буде лише у викладача.

При запуску програми для перевірки знань, перед студентом з'являється початковий екран, який дозволить вибрати режим роботи: режим тренажеру (без оцінювання) чи режим оцінювання. У випадку вибору режиму з оцінюванням з'явиться вікно, що містить поля для введення імені, прізвища, групи та ключового слова.

Далі розпочнеться виконання завдань аналогічних до завдань тренажеру [7], але при цьому буде підраховуватися кількість помилок різних типів.

Після того, як користувач завершить відповіді на питання, відкриється кінцева форма. В ній буде інформація про успішність проходження студента даної теми. А саме: процент пройдених кроків, кількість помилок різних типів, час початку і кінця, дані про студента та код, обрахований за допомогою деякої невідомої студенту хеш-функції. Інформацію із цієї форми студент має відправити викладачу.

Для перевірки викладачем результатів буде використовуватися програма для верифікації, в яку потрібно буде помістити відправлену студентом інформацію. Якщо студент змінить інформацію про успішність проходження, то хеш-код не буде співпадати і програма проінформує викладача про це.

Отже, комплекс програм розробляється для вивчення та оцінювання знань з теми «Аналіз алгоритму сортування злиттям». Програми допоможуть студенту навчитися розв'язувати завдання з даної теми, а викладачу оцінити успіхи студента. Тренажер буде використовуватися при вивченні відповідної теми дисципліни «Аналіз алгоритмів».

Література

1. Ярмоленко А. В. Алгоритм роботи тренажеру з теми «Асимптотичні оцінки функцій» дисципліни «Аналіз алгоритмів» / А. В. Ярмоленко, Ю. Ф. Олексійчук // Комп'ютерні науки і прикладна математика (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 2 – Полтава: Кафедра ММСІ ПУЕТ, 2018. – С.14-16.
2. Голубенко Вл. О. Програмна реалізація тренажеру з теми «Сортування бульбашками» дисципліни «Аналіз алгоритмів» / В. О. Голубенко, Ю. Ф. Олексійчук // Комп'ютерні науки і прикладна математика (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 2 – Полтава: Кафедра ММСІ ПУЕТ, 2018. – С. 6-9.
3. Русін В. С. Програмна реалізація елементів тренажеру з теми "Аналіз алгоритму сортування вставками" дисципліни "Аналіз алгоритмів" / В. С. Русін, Ю. Ф. Олексійчук // Інформатика та системні науки (ІСН-2017): матеріали VIII Всеукраїнської

науково-практичної конференції за міжнародною участю (м. Полтава, 16–18 березня 2017 р.) – Полтава: ПУЕТ, 2017. – С. 236-237.

4. Ємець О. О. Про розробку тренажерів для дистанційних курсів кафедрою ММСІ ПУЕТ / О.О. Ємець // Інформатика та системні науки (ІСН-2015): матеріали VI Всеукраїнської науково-практичної конференції за міжнародною участю, (м. Полтава, 19–21 берез. 2015 р.). – Полтава: ПУЕТ, 2015. – С. 152-161. – Режим доступу:

<http://dspace.puet.edu.ua/handle/123456789/2616>

5. Парфьонова Т. О. Про розробку тренажерів для дистанційного навчального курсу "Алгебра і геометрія" / Т. О. Парфьонова // Інформатика та системні науки (ІСН-2016): матеріали VII Всеукраїнської науково-практичної конференції за міжнародною участю, (м. Полтава, 10–12 берез. 2016 р.) – Полтава: ПУЕТ, 2016. – С. 221-223.

6. Чілікіна Т. В. Огляд тренажерів з дисципліни "Математичний аналіз" на прикладі розробок студентів напряму "Інформатика" / Т. В. Чілікіна // Інформатика та системні науки (ІСН-2016): матеріали VII Всеукраїнської науково-практичної конференції за міжнародною участю, (м. Полтава, 10–12 берез. 2016 р.). – Полтава: ПУЕТ, 2016. – С. 329-330.

7. Фесенко Д. І. Розробка тренажера з теми «Аналіз алгоритму сортування злиттям» дисципліни «Аналіз алгоритмів» / Д. І. Фесенко, Ю. Ф. Олексійчук // Актуальні питання розвитку науки та забезпечення якості освіти у XXI столітті: тези доповідей XLIII Міжнародної наукової студентської конференції за підсумками науково-дослідних робіт студентів за 2019 рік (м. Полтава, 07–08 квітня 2020 року). Частина 2 – Полтава: ПУЕТ, 2020 – С.129-131. – Режим доступу:

<http://dspace.puet.edu.ua/handle/123456789/8290>

8. Кормен Т. Алгоритмы: построение и анализ, 2-е изд./ Т. Кормен, Ч. Лейзерсон, Р. Ривест, К. Штайн — М.: Вильямс, 2005. — 1296 с.

СТВОРЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ З ТЕМИ «ЛОГІЧНІ ОПЕРАТОРИ PHP»

*М.М. Клименко, студент гр. КН б інт-21, спеціальності
«Комп'ютерні науки»*

*Вищий навчальний заклад Укоопспілки «Полтавський
університет економіки і торгівлі»
klymenkomykola35@gmail.com*

*Т. В. Чілікіна, к.ф.-м.н., доцент
ВНЗ Укоопспілки ПУЕТ
tv.0502@ukr.net*

*В доповіді розглядається алгоритм та створений по ньому
тренажер з теми «Логічні оператори PHP».*

*Klymenko M.M., Chilikina T.V. Creation of software on the
topic "Logical operators PHP". The algorithm and the simulator for
the topic "Logical operators of PHP" are considered in the report.*

Ключові слова: ТРЕНАЖЕР, ЛОГІЧНІ ОПЕРАТОРИ, PHP.

Keywords: SIMULATOR, LOGIC OPERATORS, PHP.

В тезах викладено постановку задачі та алгоритм роботи програмного забезпечення «Логічні оператори PHP».

Головним завданням є створення програмного забезпечення у вигляді тренажера з теми «Умовні оператори PHP».

Програмне забезпечення створено у середовищі розробки MS Visual Studio, з використанням мови програмування C#.

Головною задачею тренажера буде навчання студентів працювати з логічними операторами PHP, в цьому і полягає його актуальність.

Після проведення пошуку на електронному ресурсі dSPACE.puet.edu.ua програм-тренажерів з теми «Умовні

оператори мови PHP» було знайдено лише інформацію пов'язану з розробкою сайтів, але не програму у вигляді тренажеру для навчання конкретній темі. Саме тому вважається доцільним розробка тренажеру з теми «Умовні оператори мови PHP».

Розроблено алгоритм, який реалізовано у вигляді програмного забезпечення.

Під час роботи з тренажером використовуються питання з декількома варіантами відповідей, після вибору правильної відповіді буде можливість перейти далі, у разі вибору неправильної відповіді передбачена кнопка повернення до завдання теоретичного типу та кнопка підказки.

Під час роботи з практичними завданнями у першому випадку користувач отримує певну задачу з трьома варіантами відповіді, при виборі правильного варіанту з'являється повідомлення про правильну відповідь у вигляді «✓» та з'являється кнопка переходу до наступного завдання, при виборі неправильної відповіді з'являється повідомлення про помилку у вигляді «X» та з'являється кнопка «Підказка». Також передбачені практичні завдання з можливістю вводу свого варіанту відповіді.

Алгоритм роботи з навігаційними елементами тренажеру описано такими кроками.

Крок 1: Запуск тренажеру та вибір оптимальних налаштувань для екрану.

Крок 2: Перехід до головного меню тренажеру. В головному меню описано тему та надано доступ до елементів тренажеру – «Довідка», «Вибір практичного завдання» та «Інформація про тренажер».

Крок 3: При переході до елемента «Інформація про тренажер» користувач отримує інформацію про виконавця та керівника роботи. Для повернення до головного меню необхідно натиснути кнопку «ОК».

Крок 4: При переході до елемента «Довідка» користувач отримує доступ до кнопок для переходу до певного довідкового матеріалу тренажеру. Існує можливість повернутися до меню «Довідка» натиснувши кнопку «Тема» в довідковому матеріалі.

Крок 5: При переході до елементу «Вибір практичної теми» користувач отримує доступ до вікна з трьома практичними темами. При натисненні на практичну тему користувач отримує доступ до практичних завдань з неї. Існує можливість повернутися до меню «Вибір практичної теми» натиснувши кнопку «До практики» під час роботи з практичним завданням.

Алгоритм роботи з навчальними матеріалами описано наступними кроками:

Крок 1: Після вибору довідкового матеріалу користувач переходить до вікна з короткою довідкою по певній темі. Після ознайомлення з довідкою користувач переходить до практичних завдань по першій темі. Існує можливість обрати будь-яке практичне завдання через відповідне меню.

Крок 2: Користувач отримує практичне завдання: з теми «IF-ELSE».

Задача 1. Якщо \$a дорівнює 10 то вивести "true" якщо ні - "false".

Заповніть пропуски ("...") в коді:

```
<?php
$a = 10;
if ($a ... 10) {
    echo 'true';
} else {
    echo 'false';
}
?>
```

Варіанти (правильний 1) –

1. ==
2. !=
3. ===»

Крок 3: Користувач отримує практичне завдання:

Задача 2. Дана наступна програма, що є її результатом?

```
<?php
$a == 0;
if ($a < 0) {
    echo 'true';
} else {
```

```
echo 'false';  
}  
?>
```

Варіанти (правильний 2) –

1. true
2. false
3. error»

Крок 4: Користувач отримує практичне завдання:

Задача 3. Дана наступна програма, що є її результатом?

```
<?php  
$a = -3;  
if ($a == 0) {  
echo 'true';  
} else {  
echo 'false';  
}  
?>
```

Варіанти (правильний 2) –

1. true
2. false
3. error

Крок 5: Користувач отримує практичне завдання:

Задача 4. Дана наступна програма, що є її результатом?

```
<?php  
$a = -12;  
if ($a != 0) {  
echo 'true';  
} else {  
echo 'false';  
}  
?>
```

Варіанти (правильний 1) –

1. true
2. !true
3. False

Крок 6: Користувач отримує практичне завдання:

Задача 5. Дана наступна програма, що є її результатом?

```
<?php
$a = "abc";
if ($a === "abc") {
    echo 'true';
} else {
    echo 'false';
}
?>
```

Варіанти (правильний 3) –

1. false
2. error
3. true

Крок 7: Користувач отримує практичне завдання з наступної теми:

ЛОГІЧНІ ЗМІННІ.

Задача 1. Дано наступний код, що є результатом його виконання?

```
<?php
$a = 4;
$b = 3;
$c = 66;
$var = true;
if ($var == true) {
    echo $a;
}
else {
    echo 'sample1';
}
if ($var == false) {
    echo $b;
}
else {echo 'sample2';}
?>
```

Варіанти (правильний 1) –

1. 4sample2
2. 3sample2
3. 4sample1

Крок 8: Користувач отримує практичне завдання:

Задача 2. Дано наступний код, що є результатом його виконання?

```
<?php
$a = 4;
$b = 3;
$c = 66;
$var = !true;
if ($var == true) {echo $a;}
else {echo 'sample1';}
if ($var == false) {echo $b;}
else {echo 'sample2';}
?>
```

Варіанти (правильний 3) –

1. sample14
2. sample1sample2
3. sample13

Крок 9: Користувач отримує практичне завдання:

Задача 3. Дано наступний код, що є результатом його виконання?

```
<?php
$a = 4;
$b = 3;
$c = 66;
$var = false;
if ($var) {
    echo $a;
}
else {
    echo 'sample1';
}
if (!$var) {
    echo $b;
}
else {echo 'sample2';}
?>
```

Варіанти (правильний 2) –

1. sample14
2. sample13
3. sample1sample2

Крок 10: Користувач отримує практичне завдання:

Задача 4. Дано наступний код, що є результатом його виконання?

```
<?php
$a = 4;
$b = 3;
$c = 66;
$var = false;
if ($var == true) {
    echo $a;
}
else {
    echo $c;
}
if ($var == false) {
    echo $b;
}
else {echo 'sample2';}
?>
```

Варіанти (правильний 2) –

1. 664
2. 663
3. 66sample2

Крок 11: Користувач отримує практичне завдання:

Задача 5. Дано наступний код, що є результатом його виконання?

Введіть правильну відповідь у відповідному полі (оберіть мову "Англійська")

```
<?php
$a = 4;
$b = 3;
$c = 66;
$var = !false;
if ($var == !true) {
```

```

echo $a;
}
else {
echo $c;
}
if (!$var) {
echo $b;
}
else {echo 'sample2';}
?>

```

Після вводу правильна відповідь «66sample2»

Крок 12: Користувач отримує практичне завдання з наступної теми:

SWITCH-CASE.

Задача 1. Сайт підтримує 3 мови - російська, англійська, німецька. Змінна \$ lang може приймати 3 значення - 'ru', 'en' і 'de'. Залежно від значення змінної \$ lang слід вивести фразу на одній з мов.

Заповнити пропуски в коді.

```

<?php
switch ($lang) {
case '??':
echo 'Російська';
break;
case '??':
echo 'Англійська';
break;
case '??':
echo 'Німецька';
break;
default:
echo 'Ця мова не підтримується';
break;
}
?>

```

Варіанти відповіді (Правильна 1) :

1. ru en de

2. en ru de

3. ru de en

Крок 13: Користувач отримує практичне завдання:

Задача 2. Змінна \$ l може приймати 2 значення: 'ru' і 'en'.

Якщо вона має значення 'ru', то в змінну \$ a запишемо масив днів тижня російською мовою, а якщо має значення 'en' - то англійською.

Заповніть пропуски в коді.

```
<?
$l='ru';
switch($l){
case 'ru':
$a=['пн','вт','ср','чт','пт','сб','вс'];
break;
case 'en':
$a=['mn','ts','wd','th','fr','st','sn'];
break;
}
????
?>
```

Варіанти відповіді (Правильна 3) :

1. printf a
2. echo \$a
3. var_dump(\$a)

Крок 14: Користувач отримує практичне завдання:

Задача 3. Який результат коду, що наведено нижче?

```
<? Php
$ Speed = 55;
switch ($ speed)
{
case 30:
echo "Ваша швидкість 30 км / год";
break;
case 58:
echo "Ваша швидкість 50 км / год";
break;
case 70:
```

```
echo "Перевищення швидкості!";  
break;  
default:  
echo "Швидкість в межах норми";  
break;  
}  
?>
```

Варіанти відповіді (Правильна 2) :

1. Перевищення швидкості!
2. Швидкість в межах норми
3. Ваша швидкість 55 км / год

Крок 15: Після вибору правильного варіанту відповіді користувач отримує доступ до кнопки «Наступне завдання»

Крок 16: Після вибору неправильного варіанту відповіді користувач отримує доступ до кнопки «Підказка», що надає коротку підказку до завдання та доступ до кнопки, що дозволяє ознайомитися з теорією до завдання. Після переходу до теорії, з'являється кнопка повернення до практичного завдання.

Створено програмне забезпечення у вигляді тренажеру з теми «Логічні оператори РНР» надасть можливість студентам опрацювати і закріпити знання з даної теми, планується впровадити його навчальний процес.

Література

1. Трепачёв Д. Учебник РНР для новичков от Трепачёва Дмитрия / Дмитрий Трепачёв. – Режим доступа до ресурсу: <http://old.code.mu/books/php/base/rabota-s-konstrukciyami-if-else-switch-case-v-php.html>

**СТВОРЕННЯ ТРЕНАЖЕРУ З ТЕМИ
«КОДИ ІЗ ВИПРАВЛЕННЯМ ПОМИЛОК»**

О. В. Купченко, студент

Полтавський університет економіки і торгівлі

kupchenko1999@gmail.com

Т. О. Парфьонова, доцент, канд.фіз.-мат. наук

Полтавський університет економіки і торгівлі

tpa.poltava@gmail.com

В статті розглядається розробка тренажеру з теми «Коди із виправленням помилок».

Kupchenko O. V., Parfonova T. O. Creating a simulator on the topic "Error Correction Codes". In the abstracts of the report the simulator development on the topic "Error correction codes" are discussed.

Ключові слова: ТРЕНАЖЕР, КОД, ВИПРАВЛЕННЯ ПОМИЛОК.

Keywords: SIMULATOR, CODE, ERROR CORRECTION.

Коди із виправленням помилок використовуються на різних рівнях роботи з інформацією, в зв'язку з тим, що в процесі зберігання даних та передачі інформації по мережам зв'язку неминуче виникають помилки. Різноманітні області використання виправлення помилок диктують різні вимоги до використовуваних стратегій та кодів.

Все це збільшує кількість різноманітних задач і запитань, які використовуються для сприяння засвоєння матеріалу для студентів вищих навчальних закладів та фахівців з телекомунікацій, обчислювальної техніки, автоматизованих систем.

Створюваний тренажер сприяє спрощенню вивчення матеріалу з теми «Коди із виправленням помилок», а саме: створенню тестових завдань для перевірки знань теоретичного матеріалу студентами вищих навчальних закладів та фахівцями з телекомунікацій, обчислювальної техніки та автоматизованих систем. Також за допомогою тренажеру пришвидшено вивчення принципів та способів кодування повідомлень за допомогою кодів, що виправляють помилки. Використовуючи тренажер надається можливість вирішення різноманітних задач по використанню методів кодування із виправленням помилок.

Допоміжними елементами тренажеру є: можливість створення тестових завдань та задач, а також – наглядне кодування і декодування повідомлень, використовуючи різні методи кодування повідомлень з виправленням помилок. Головною особливістю наглядного кодування і декодування повідомлень є можливість імітації передачі повідомлення каналом, в якому можуть виникати помилки.

В результаті було створено тренажер використовуючи основні принципи теми «Коди із виправленням помилок», а саме: принципи кодування повідомлень, способи коректувального кодування двійковими та недвійковими кодами, методику розрахунку завадостійкості кодованих повідомлень.

Література

1. Жураковський Ю.П. Теорія інформації та кодування: Підручник / Ю.П. Жураковський, В.П. Полторак – К.: Вища шк., 2001. – 255 с.
2. Блейхут Р. Теория и практика кодов, контролирующих ошибки: Пер. с англ. / Р. Блейхут – М.: Мир, 1986. – 576 с.

**РОЗРОБКА ЕЛЕМЕНТІВ ТРЕНАЖЕРУ З ТЕМИ
«СОРТУВАННЯ ВКЛЮЧЕННЯМИ З ЛІНІЙНИМ ТА
БІНАРНИМ ПОШУКОМ» ДИСТАНЦІЙНОГО
НАВЧАЛЬНОГО КУРСУ «АЛГОРИТМИ ТА СТРУКТУРИ
ДАНИХ» ТА РОЗРОБКА ЙОГО ПРОГРАМНОГО
ЗАБЕЗПЕЧЕННЯ.**

В. В. Олефіренко, студент групи КН м-21
Полтавський університет економіки і торгівлі

В статті розглядається розробка елементів тренажеру з теми «Сортування включеннями з лінійним та бінарним пошуком» дистанційного навчального курсу «Алгоритми та структури даних» та розробка його програмного забезпечення.

Olefirenko V. V., Yemets O. O. Development of simulator elements with the same "Sorting included with linear and binary search" remote basic course "Algorithms and data structures" and analysis of its programmable software.

Ключові слова: ТРЕНАЖЕР, СОРТУВАННЯ ВКЛЮЧЕННЯМИ, ЛІНІЙНЕ СОРТУВАННЯ, БІНАРНЕ СОРТУВАННЯ, WINDOWS FORM, C#.

Keywords: SIMULATOR, INSERTION SORT, LINEAR INSERTION SORT, BINARY INSERTION SORT, DECISION TREE, WINDOWS FORM, C#.

Оскільки останнім часом в зв'язку з пандемією дистанційна освіта стала одним із головних напрямів розвитку та надання освіти тому на даний момент часу створюють велику кількість різних методів надання інформації та для навчання учнів. Застосування в навчанні комп'ютерів значно спрощує надання інформації учням які не мають змоги відвідувати навчальні заклади.

У зв'язку з цим, в останній час масово розробляють різні методи навчання на відстані, які допомагають викладачеві

краще надавати інформацію для студентів.

Принцип роботи тренажера з теми «Сортування включеннями з лінійним та бінарним пошуком» дистанційного навчального курсу «Алгоритми та структури даних»: користувачу надаються різні питання як теоретичні так і практичні, необхідно дати відповідь на питання в відповідні поля тренажера

Для створення питань використовується дистанційний курс «Алгоритми та структури даних»[3]. Сортування включеннями з лінійним та бінарним пошуком впорядковує елементи масива.

Тренажер розроблено на мові програмування C# в інтегрованому середовищі розробки Visual Studio.

Призначення тренажера:

- 1) підготовка студентів;
- 2) спрощення опрацювання навчального матеріалу.

Основними перевагами розробленого тренажера є:

- 1) Зручний інтерфейс.
- 2) Можливість інтегрування в дистанційний навчальний курс.
- 3) Не потребує доступу в мережу інтернет для роботи з ним.

Тренажер можливо використовувати для вивчення теми «Сортування включеннями з лінійним та бінарним пошуком» дистанційного навчального курсу «Алгоритми та структури даних».

Література

1. Ємець О. О. Методичні рекомендації щодо оформлення пояснювальних записок до курсових проектів (робіт) для студентів за освітньою програмою «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки та інформаційні технології», «Комп'ютерні науки» галузь знань - 12 «Інформаційні технології» / О. О. Ємець – Полтава : РВВ ПУЕТ, 2017. – 69с.
2. Панішев А.В. Вступ до теорії складності дискретних задач : Монографія. / А.В. Панішев – Ж. : ЖДТУ, 2004. – 236с.
3. Ємець О. О. Дистанційний курс ПУЕТ «Алгоритми та

структури даних» для студентів спеціальностей «Комп'ютерні науки та інформаційні технології/ О. О. Ємець. – [Електронний ресурс].

4. Алгоритм сортування [Електронний ресурс] / Матеріал з Вікіпедії — вільної енциклопедії. Режим доступу: https://uk.wikipedia.org/wiki/Алгоритм_сортування
5. Алгоритми сортування порівняння [Електронний ресурс]. Режим доступу: <http://www.cs.usfca.edu/~galles/visualization/ComparisonSort.html>
6. Сортування включенням [Електронний ресурс] / Матеріал з Вікіпедії — вільної енциклопедії. Режим доступу: https://uk.wikipedia.org/wiki/Сортування_включенням
7. Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання: ДСТУ 7.1-2006. – [Чинний від 2007-07-01]. – К. : Держспоживстандарт України, 2007. – 47 с

ТРЕНАЖЕР
«ПОБУДОВА БЛОК-СХЕМ АЛГОРИТМІВ ЦИКЛІЧНОЇ
СТРУКТУРИ НА ПРИКЛАДІ ЦИКЛУ WHILE»

Б. М. Бибка, бакалавр спеціальності «Комп'ютерні науки»

О. О. Ємець, канд. фіз.-мат. наук, доц.

Полтавський університет економіки і торгівлі

Розглядається алгоритм тренажеру.

Bibka B. M., Yemets` O. O. Simulator «Construction of flowcharts of loop structure algorithms on the example of the while loop». The algorithm of the simulator is considered.

Ключові слова: АЛГОРИТМ, БЛОК-СХЕМА, ТРЕНАЖЕР, ЦИКЛ.

Keywords: ALGORITHM, FLOWCHART, SIMULATOR, LOOP.

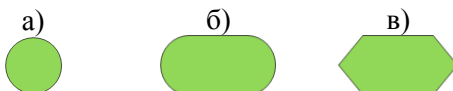
Умова завдання. Побудувати блок-схему алгоритму:

```
const
  N = 5;
var
  a, Sum, i: integer;
BEGIN
  Sum := 0;
  i := 1;
  while (i <= N) do
  begin
    write('Введіть число: ');
    readln(a);
    Sum := Sum + a;
    i := i + 1;
  end;
  writeln('Сума =', Sum);
  readln;
END.
```


Якщо відповідь вірна, то відбувається перехід до фрагменту блок-схеми. У випадку помилки – з'являється її пояснення. Користувач повинен виправити похибку.

У випадку відсутності відповіді на екрані з'являється попереджуваче повідомлення з вказівкою обрати відповідь.

Крок 1. Початок алгоритму відображається символом



Вірна відповідь – б. Якщо відповідь хибна, то з'являється повідомлення: «Початок, кінець або зупинка алгоритму відображаються символом «термінатор», який має форму овалу зі сплюснутими сторонами.»

На екрані з'являється рис. 1:

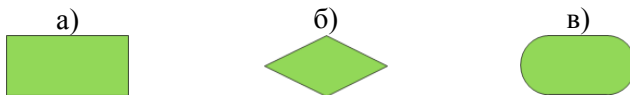


Рис. 1 – Фрагмент блок-схеми

Крок 2. Рядок

$N = 5;$

відображається символом:



Вірно – а. При похибці: «Усі маніпуляції з даними відображаються у символі «процес», який має форму прямокутника.»

З'являється рис. 2:

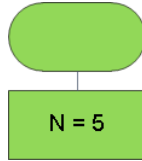
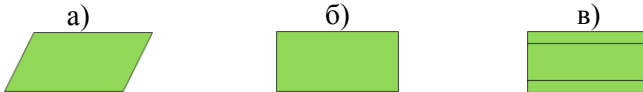


Рис. 2 – Фрагмент блок-схеми

Крок 3. Рядок

`Sum := 0;`

відображається символом:



Вірно – б. При похибці: «Усі маніпуляції з даними відображаються у символі «процес», який має форму прямокутника без смуг.».

З'являється рис. 3:

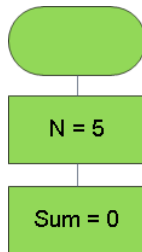
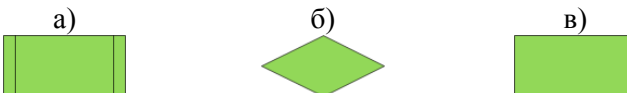


Рис. 3 – Фрагмент блок-схеми

Крок 4. Рядок

`i := 1;`

відображається символом:



Вірно – в. При похибці: «Усі маніпуляції з даними відображаються у символі «процес», який має форму прямокутника без смуг.».

З'являється рис. 4:

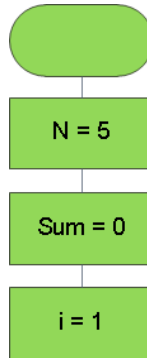
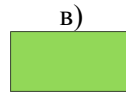
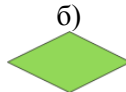
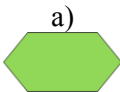


Рис. 4 – Фрагмент блок-схеми

Крок 5. Рядок

`while (i <= N) do`
відображається символом:

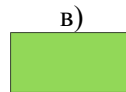
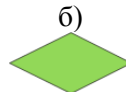
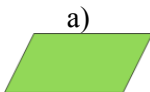


Вірно – б. При похибці: «Перевірка умови циклу відображається символом «рішення», який має форму ромба.».

З'являється рис. 5.

Крок 6. Рядок

`write('Введіть число: ');`
відображається символом:



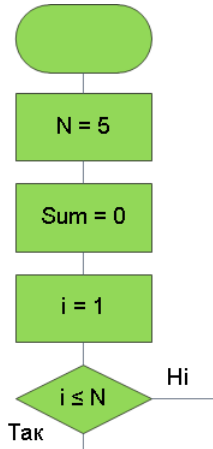


Рис. 5 – Фрагмент блок-схеми

Вірно – а. При похибці: «Для виводу інформації використовується символ «дані», який має форму паралелограма.».

З'являється рис. 6:

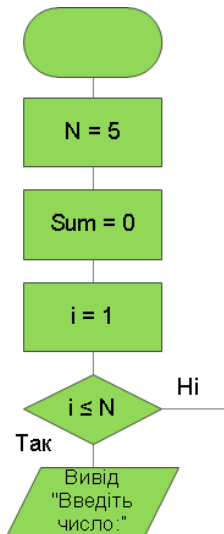
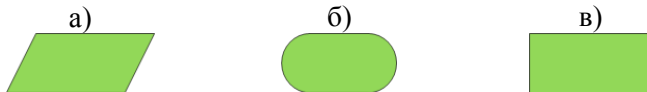


Рис. 6 – Фрагмент блок-схеми

Крок 7. Рядок

`readln(a);`

відображається символом:



Вірно – а. При похибці: «Для вводу інформації (як і для виводу) використовується символом «дані», який має форму паралелограма.».

З'являється рис. 7.

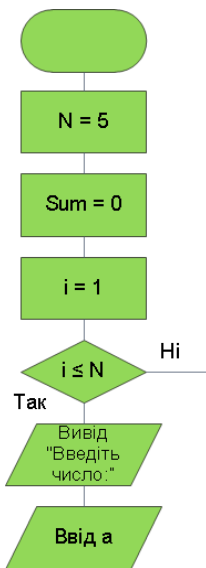
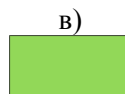
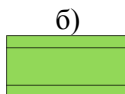
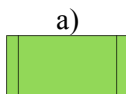


Рис.7 – Фрагмент блок-схеми

Крок 8. Рядок

`Sum := Sum + a;`

відображається символом:



Вірно – в. При похибці: «Усі арифметичні обчислення відображаються у символі «процес», який має форму прямокутника без смуг.».

З'являється рис. 8.

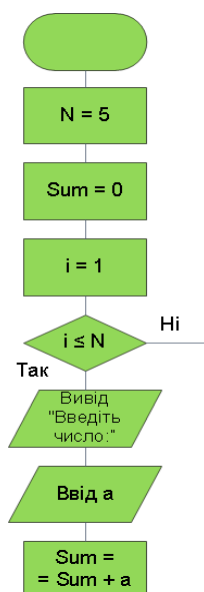
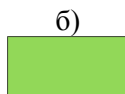
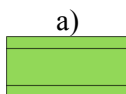


Рис. 8 – Фрагмент блок-схеми

Крок 9. Рядок

$i := i + 1;$

відображається символом:



Вірно – б. При похибці: «Усі арифметичні обчислення відображаються у символі «процес», який має форму прямокутника без смуг.».

З'являється рис. 9:

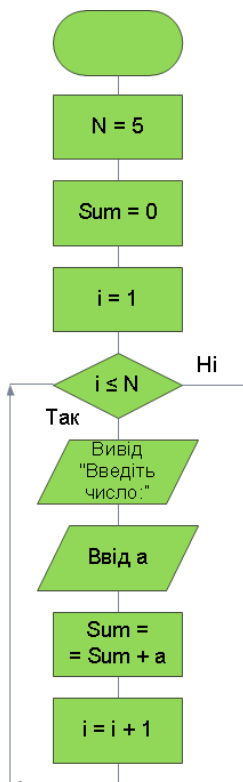
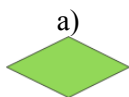


Рис. 9 – Фрагмент блок-схеми

Крок 10. Рядок

`writeln('сума =', Sum);`

відображається символом:



Вірно – в. При похибці: «Для виводу інформації використовується символ «дані», який має форму паралелограма.».

З'являється рис. 10.

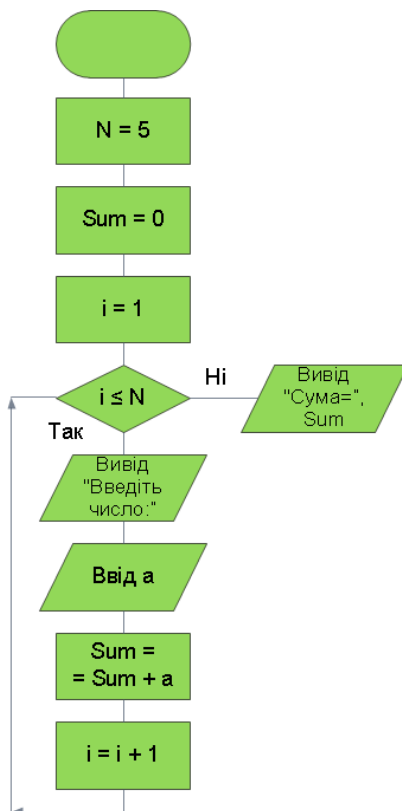
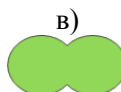


Рис. 10 – Фрагмент блок-схеми

Крок 11. Кінець алгоритму відображається символом



Вірно – б. При похибці: «Початок, кінець або зупинка алгоритму відображаються символом «термінатор», який має форму овалу зі сплюснутими сторонами.»

З'являється результуючий рис. 11.

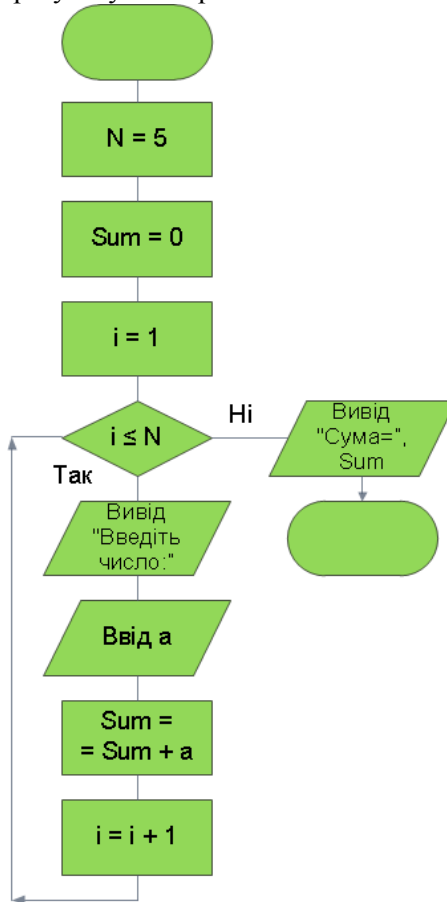


Рис. 11 – Кінцева блок-схема

Література

1. Ємець О. О. Про розробку тренажерів для дистанційних курсів кафедри ММСІ ПУЕТ / О. О. Ємець // Інформатика та системні науки (ІСН-2015): матеріали VI Всеукр. наук.-практ.

- конф. за міжн. участю (м. Полтава, 19-21 березня 2015 р.) / за ред. Ємця О. О. – Полтава: ПУЕТ, 2015. – С. 152-161. – Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/2488>.
2. Гмиза Б. Ю. Тренажер з теми «Побудова блок-схем алгоритмів циклічної структури на прикладі циклу `for`» дистанційного навчального курсу «Інформатика» та розробка його програмного забезпечення / Б. Ю. Гмиза, Ол-ра О. Ємець // Комп'ютерні науки і прикладна математика (КНіПМ-2019): матеріали наук.-практ. семінару. Випуск 3. / За ред. Ємця О. О. – Полтава: Кафедра ММСІ ПУЕТ, 2019. – С. 38-39. – Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/7036>.
3. Мордасова І. В. Тренажер з теми «Побудова блок-схем алгоритмів розгалуженої структури» дистанційного навчального курсу «Інформатика» та розробка його програмного забезпечення / І. В. Мордасова, Ол-ра О. Ємець // Комп'ютерні науки і прикладна математика (КНіПМ-2019): матеріали наук.-практ. семінару. Випуск 3. / За ред. Ємця О. О. – Полтава: Кафедра ММСІ ПУЕТ, 2019. – С. 35-37. – Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/7037>.
4. Сузанська А. О. Тренажер «Побудова блок-схем алгоритмів розгалуженої структури» / А. О. Сузанська, Є. М. Ємець, О. О. Ємець // Комп'ютерні науки і прикладна математика (КНіПМ-2020): матеріали наук.-практ. семінару. Випуск 5. / За ред. Ємця О. О. – Полтава: Кафедра ММСІ ПУЕТ, 2019. – С. 56-61.
– Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/8906>.
5. Шакуро В. Є. Розробка програмного забезпечення з теми «Побудова блок-схема алгоритмів лінійної структури» дистанційного курсу «Інформатика» / В. Є. Шакуро, О. О. Ємець // Комп'ютерні науки і прикладна математика (КНіПМ-2019): матеріали наук.-практ. семінару. Випуск 3. / За ред. Ємця О. О. – Полтава: Кафедра ММСІ ПУЕТ, 2019. – С. 40-42.
– Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/7038>.
6. Ємець О. О. Дистанційний курс ПУЕТ «Інформатика. Частина 1» для студентів спеціальності «Комп'ютерні науки» / О. О. Ємець. – [Електронний ресурс].

**СТВОРЕННЯ ТРЕНАЖЕРУ ДИСТАНЦІЙНОГО
НАВЧАЛЬНОГО КУРСУ «ДИСКРЕТНА МАТЕМАТИКА»
З ТЕМИ «АЛГЕБРА ЖЕГАЛКІНА, СПОСОБИ ПОБУДОВИ
ПОЛІНОМІВ ЖЕГАЛКІНА»**

А. О. Бурко, студент

*Полтавський університет економіки і торгівлі
andrey.burko24@gmail.com*

*Запропоновано програмну реалізацію тренажеру з теми
«Алгебра Жегалкіна, способи побудови поліномів Жегалкіна»
дистанційного навчального курсу «Дискретна математика».*

*Burko A.O. Creation of a simulator of the distance learning
course "Discrete Math" on the topic "Zhegalkin algebra, methods of
constructing Zhegalkin polynomials"*

*The software implementation of the simulator on the topic
"Zhegalkin algebra, methods for constructing Zhegalkin
polynomials" of the distance learning course "Discrete Math" is
proposed.*

Ключові слова: ТРЕНАЖЕР, АЛГЕБРА ЖЕГАЛКІНА,
ПОБУДОВА ПОЛІНОМІВ ЖЕГАЛКІНА.

Keywords: SIMULATOR, ZHEGALKIN ALGEBRA,
CONSTRUCTING ZHEGALKIN POLYNOMIALS.

Основною мета - це створити тренажер для навчання студентів дистанційного навчального курсу «Дискретна математика» темі «Алгебра Жегалкіна, способи побудови поліномів Жегалкіна».

Об'єктом розробки є створення тренажеру для систем дистанційного навчання.

Предмет розробки — програма для елементів тренажеру для навчання темі «Алгебра Жегалкіна, способи побудови поліномів Жегалкіна».

Перелік використаних методів – було використано платформу Юніті та середовище розробки Visual Studio.

При виконанні роботи були розглянуті деякі існуючі в ПУЕТ тренажери з різних дисциплін.

Загальні основні вади оглянутих робіт:

- перехід по кроках алгоритму не потребує ніяких дій користувача;
- від користувача лише потребується натискати кнопку для переходу до наступного кроку;
- незрозумілий або незручний інтерфейс;
- для початку роботи потребує завантаження з мережі Інтернет.

Позитивні аспекти при реалізації тренажера:

- зрозумілий та зручний інтерфейс;
- потребує відповідати на поставленні питання;
- потребує розрахувати результат та ввести його у відповідне поле;
- реалізовано перехід між питаннями;
- виведення повідомлення про помилку;
- реалізує роботу користувачів в двох представленнях: викладач та студент;
- при введенні даних з'являються підказки, які пояснюють, що вводиться в те чи інше поле;
- наявна довідкова інформація, яка містить інструкцію з використання.

Інформація в тренажері видається наступним чином: спочатку виводиться коротка довідка, потім практичне завдання.

При правильній відповіді користувач отримує повідомлення «Правильна відповідь», у іншому випадку користувач отримує повідомлення «Неправильна відповідь» та правильну відповідь.

Переміщення між частинами завдання реалізовані у вигляді кнопок «Далі» та «Назад».

Після закінчення роботи з тренажером з'явиться кнопка повтору, що перенаправить користувача на початок роботи з тренажером.

Результати:

1. Обрано програмне середовище –Visual Studio та мову програмування C#;
2. Програмний продукт розроблено на платформі Юніті;
3. Розроблено алгоритм роботи тренажера з теми «Алгебра Жегалкіна, способи побудови поліномів Жегалкіна» для дистанційного курсу «Дискретна математика».
4. Складено блок-схеми до алгоритму.
5. Програмно реалізовано навчальну програму.
6. Виконано тестування програми.

Плюсами розробленого програмного забезпечення є:

1. Приємний очам дизайн та інтуїтивно зрозумілий інтерфейс програми.
2. Передбачено перевірку на правильність вводу даних та вивід відповідного повідомлення.

Література

1. Яблонский С. В. Введение в дискретную математику: Учеб. Пособие для вузов. - 2-е изд., перераб. и доп. / С.В. Яблонский – М.: Наука, 1986. – 384 с.
2. Марченков С. С. Замкнутые классы булевых функций./ С.С. Марченков - М.: Физматлит. - 2001. - 128 с.
3. Супрун В.П. Табличный метод полиномиального разложения булевых функций / В.П. Супрун // Кибернетика. –1987. – № 1. – С. 116 – 117.
4. Ємець Олег Олексійович. Дискретна математика: Навчальний посібник. Видання 2-ге, доповнене / О. О. Ємець, Т. В. Парфьонова. – Полтава: РВВ ПУСКУ, 2009. –287 с.
Режим доступу до посібника:
<http://dspace.puet.edu.ua/handle/123456789/552>

**РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ
ТРЕНАЖЕРА З ТЕМИ «ТЕОРЕМИ ДОДАВАННЯ ТА
МНОЖЕННЯ ЙМОВІРНОСТЕЙ ВИПАДКОВИХ ПОДІЙ»
ДИСТАНЦІЙНОГО НАВЧАЛЬНОГО КУРСУ «ТЕОРІЯ
ЙМОВІРНОСТЕЙ І МАТЕМАТИЧНА СТАТИСТИКА»**

*Д. А. Дудник, студент гр. КН б-41, спеціальності
«Комп'ютерні науки»*

*Вищий навчальний заклад Укоопспілки «Полтавський
університет економіки і торгівлі»
dudnik.dmitriy.2000@gmail.com*

Т. О. Парфьонова, доцент, канд. фіз.-мат. наук

*Вищий навчальний заклад Укоопспілки «Полтавський
університет економіки і торгівлі»
tapa.poltava@gmail.com*

У доповіді розглядається алгоритм роботи програмного забезпечення для навчального тренажера. Він базується на покроковому розв'язку задачі про знаходження ймовірності суми несумісних подій.

Dudnik D., Parfyonova T. Development of the training simulator's software on the topic of "Addition and multiplication theorems of probability" for the distance training course "Probability theory and mathematical statistics". The algorithm of the training simulator's software is considered in the publication. The algorithm is based on the step-by-step solving of the problem about addition theorems of the incompatible events.

Ключові слова: НАВЧАЛЬНИЙ ТРЕНАЖЕР, ТЕОРІЯ ЙМОВІРНОСТЕЙ, НЕСУМІСНІ ПОДІЇ

Keywords: TRAINING SIMULATOR, PROBABILITY THEORY, INCOMPATIBLE EVENTS.

Навчальний тренажер – нині актуальне програмне забезпечення для учбового процесу університетів, дистанційних курсів та інших закладів освіти, яке підвищує якість самостійної підготовки учнів.

Досліджуючи вже наявні тренажери на дистанційному курсі було виявлена відсутність навчального тренажера з теми «Теореми додавання та множення ймовірностей випадкових подій» для дистанційного навчального курсу «Теорія ймовірностей і математична статистика». Можна відмітити несистематичний підхід до структури та оформлення деяких тренажерів. Ці факти породжують потребу у розробці нового навчального тренажера.

Далі наведена частина алгоритму однієї із задач, що реалізовані в тренажері на тему «Теореми додавання та множення ймовірностей випадкових подій» дистанційного навчального курсу «Теорія ймовірностей і математична статистика».

На стартовому вікні знаходиться інформація про: університет, виконавця та наукового керівника. Також наведена версія програми.

Посередині стартового вікна знаходяться кнопки, що ведуть до відповідних розділів:

- Задачі про пошук ймовірності суми подій,
 - Задачі про пошук ймовірності добутку подій,
 - Задачі про пошук ймовірності комбінуванням теорем суми та добутку подій.
- У розділі «Задачі про пошук ймовірності суми подій» користувачеві пропонується вибрати між розділами:
- Знаходження ймовірності суми несумісних подій,
 - Знаходження ймовірності суми сумісних подій.

Після чого користувач розпочне покроковий інтерактивний розв'язок вибраної задачі.

Інтерфейс цього розділу складається із:

- закріпленого зверху блоку з умовою задачі,
- блоку з кроками розв'язування задачі.

На кожному кроці користувачеві пропонується дати відповідь на тестове завдання або записати свої обчислення задачі у відповідні поля.

Частина кроків може бути супроводжена додатковими поясненнями та відомостями.

Перевірка відповіді відбувається у режимі реального часу – як тільки користувач вибере варіант відповіді або заповнить усі поля прикладу. Для навігації в полях рекомендується використовувати кнопку «Tab».

Якщо відповідь правильна, то вона візуально відмічається зеленим кольором та даний блок стає недоступним для зміни. Після чого здійснюється перехід до наступного кроку.

Якщо відповідь неправильна, то вона візуально відмічається червоним кольором та по центру програми з'являється вікно, яке повідомляє про помилку та наводить підказку до правильної відповіді.

Розглянемо декілька кроків алгоритму розв'язку задачі про знаходження ймовірності суми несумісних подій. В алгоритмі усього 17 кроків.

Далі в описі алгоритму в тестових запитаннях лише перша відповідь правильна. В тренажері відповіді перемішуються.

Умова задачі: В урни знаходиться 7 червоних та 3 чорних кулі. 3 ящика виймають навмання 2 кулі. Яка ймовірність того, що вони однокольорові?

Крок 1: Нехай:

Подія А з урни витягли 2 червоні кулі

Подія В з урни витягли 2 чорних кулі

Подія М - витягнуто однокольорові кулі

Згадаємо деякі теоретичні відомості.

Випадкові події А та В називаються сумісними, якщо:

- поява однієї події не виключає

- поява однієї події виключає появу іншої.
- поява однієї події залежить від появи іншої.
- поява однієї події відбувається лише одночасно з появою іншої.

Крок 6: Ймовірність суми несумісних подій:

$$\begin{array}{l} \cdot \quad P(A + B) = P(A) + P(B) \\ \cdot \quad P(A + B) = P(A) * P(B) \\ \cdot \quad P(A + B) = P(A) + P(B) - P(A * B) \\ \cdot \quad P(A + B) = P(A + B) - P(A * B) \end{array}$$

Крок 8: Число загальних випадків – число всіх рівноможливих і несумісних випадків, тобто кількість пар які можна утворити з 10 куль.

- Кількість сполучень C_n^m
- Кількість перестановок A_m^n
- Кількість розміщень P_n

Крок 9: Яка формула використовується для обчислення кількості сполучення?

$$C_n^m = \frac{n!}{m!(n-m)!}$$

$$C_n^m = \frac{n!}{n!(n-m)!}$$

$$C_n^m = \frac{m!}{m!(n-m)!}$$

$$C_n^m = \frac{n!}{m!(m-n)!}$$

Якщо у цьому кроці користувач вибере неправильну відповідь, то з'явиться вікно із текстом «для обчислення кількості сполучення використовується формула

$$C_n^m = \frac{n!}{m!(n-m)!} . \gg$$

Крок 10: Отже, число загальних випадків:

$$C_{10}^2 = \frac{\quad \times \quad}{\quad} = \quad$$

Якщо у цьому кроці користувач введе правильні дані у ці поля, що матимуть вигляд:

$$C_{10}^2 = \frac{9 \cdot 10}{2} = 45 ,$$

то відповідь зарахується та користувач перейде до наступного кроку.

Якщо відповідь неправильна, то з'явиться вікно із текстом «для обчислення кількості сполучення використайте

$$\text{формулу } C_n^m = \frac{n!}{m!(n-m)!} . \gg$$

В публікації розглянуті лише деякі кроки алгоритму тренажера з теми «Теорема додавання та множення ймовірностей випадкових подій»

Розроблений тренажер реалізує ряд задач, та може бути впроваджений у навчальний процес.

Література

1. Огірко О. І. Теорія ймовірностей та математична статистика: навчальний посібник / О. І. Огірко, Н. В. Галайко. – Львів: ЛьвДУВС, 2017. – 292 с.
2. Ємець О.О. Теорія ймовірностей і математична статистика. Конспект лекцій. Ч.1. Елементи теорії ймовірностей.: навч. Посібник / Укл. Ємець. – Полтава, 2001. – 34 с.

УДК 004.4'2

**ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ТРЕНАЖЕРУ З ТЕМИ
«ЛЯМБДА-ВИРАХУВАННЯ І ТИПІЗАЦІЯ»
ДИСТАНЦІЙНОГО НАВЧАЛЬНОГО КУРСУ «ТЕОРІЯ
ПРОГРАМУВАННЯ»**

*Закіров Р.Р., бакалавр спеціальності «Комп'ютерні науки»
Вищий навчальний заклад Укоопспілки «Полтавський
університет економіки і торгівлі»
Zakirovmail@gmail.com*

*Запропоновано програмну реалізацію тренажеру з теми
«Лямбда-вирахування і типізація» дистанційного навчального
курсу «Теорія програмування».*

*Zakirov R.R.. The software implementation of the simulator on
the topic "Lambda subtraction and typing" of the distance learning
course "Programming Theory" is proposed.*

Ключові слова: ТРЕНАЖЕР, ЛЯМБДА-ВИРАХУВАННЯ І
ТИПІЗАЦІЯ, ТЕОРІЯ ПРОГРАМУВАННЯ.

Keywords: SIMULATOR, LAMBDA SUBTRACTION AND
TYPING, PROGRAMMING THEORY.

Основною метою є створення елементів програмного забезпечення тренажеру з теми «Лямбда-вирахування і типізація» дистанційного навчального курсу «Теорія програмування».

Об'єктом розробки є процес дистанційного навчання математичним дисциплінам.

Предмет розробки — алгоритм роботи тренажеру з теми «Лямбда-вирахування і типізація».

Перелік використаних методів – застосування математичних основ теорії програмування з теми «Лямбда-вирахування і типізація».

Розглянемо основні завдання роботи:

- описати основні вимоги до тренажеру;
- розглянути актуальність використання дистанційного навчання;
- навести теоретичний матеріал з теми для його використання в тренажері;
- розробити алгоритм тренажеру;
- скласти блок-схему розробленого алгоритму;
- описати причини вибору мови програмування для реалізації тренажеру;
- описати процес реалізації тренажеру;
- розробити тренажер з даної теми.

На стартовій сторінці пропонується переглянути теоретичний матеріал за темою або завантажити його. Також виводиться наступна інформація:

- тема тренажеру;
- назва дисципліни;
- прізвище, ініціали розробника;
- прізвище, ініціали керівника.

На кожному кроці виводиться завдання та варіанти відповіді, серед яких слід вибрати одну правильну. При неправильній відповіді відображається повідомлення про помилку, що вказує на вірну відповідь.

На останньому кроці відображається перелік питань, в якому вказано було вибрано правильну відповідь чи ні. Пропонується закрити тренажер або перейти на стартову сторінку, де можна розпочати тестування спочатку.

Безумовно, як і в кожній формі отримання знань, у дистанційній є і свої недоліки, але їх подолання стає можливим завдяки рокам практичного застосування цієї форми не лише як допоміжної та однієї з побічних, а як можливо рівної класичній формі здобуття освіти.

Зважаючи на викладене вище, ми можемо спрогнозувати певні тенденції розвитку дистанційного навчання, такі як збільшення кількості масових відкритих дистанційних курсів, розробка програм дистанційного навчання, інтеграція інтелектуальних комп'ютерних технологій у навчальний процес

дистанційної освіти, комбінування переваг дистанційного навчання із класичною формою освіти, моніторинг досягнень вищих навчальних закладів не лише в межах України, а і в усьому світі і подальше використання корисного досвіду.

Всі вимоги, описані в постановці задачі, були виконані.

Література

1. Дистанційне навчання у вищих навчальних закладах України : [Інформаційні матеріали] / МОН України. – Хмельницький : ХНУ, 2009. – 50 с.
2. Про Основні засади розвитку інформаційного суспільства в Україні на 2007-2015 роки : Закон України від 9.01.2007 № 537-V [Електронний ресурс]. – Режим доступу: <http://zakon.rada.gov.ua/cgi-bin/laws/main.cgi?nreg=537-16>.
3. Офіційні матеріали наради-семінару з питань нормативного забезпечення дистанційної форми навчання в Україні ; Національний технічний університет України «КПІ», м. Київ, 2012 [Електронний ресурс]. – Режим доступу: <http://ipo.kpi.ua/ua/distance/dlabout.html>
4. Черненко О.О. Електронний навчально-методичний посібник для самостійного вивчення навчальної дисципліни «Теорія програмування» для студентів напряму 6.040302 «Інформатика» – Режим доступу: <http://tprogr.ho.ua>.
5. Бабій М.С. Теорія програмування: Навчальний посібник [Електронний ресурс] / М.С. Бабій, О.П. Чекалов.– Суми: Вид-во СумДУ, 2009. – 181 с.

**ПРОГРАМУВАННЯ ЕЛЕМЕНТІВ ТРЕНАЖЕРА
«ГРАМАТИКИ. МОВИ, ЩО ЗАДАЮТЬСЯ
ГРАМАТИКАМИ» ДИСТАНЦІЙНОГО НАВЧАЛЬНОГО
КУРСУ «ТЕОРІЯ ПРОГРАМУВАННЯ»**

*Іжєвський Д.О., бакалавр спеціальності «Комп'ютерні науки»
Вищий навчальний заклад Укоопспілки «Полтавський
університет економіки і торгівлі»*

*Запропоновано програмування елементів тренажера
«Граматики. Мови, що задаються граматиками»
дистанційного навчального курсу «Теорія програмування».*

Ключові слова: ТРЕНАЖЕР, ГРАМАТИКИ, МОВИ.

*Izhevsky D.O. Programming of elements of the simulator
"Grammars. Languages set by grammars "of the distance learning
course" Programming Theory "*

*The programming of the elements of the simulator "Grammar.
Languages set by grammars "of the distance learning course"
Programming Theory " is offered.*

Keywords: SIMULATOR, GRAMMARS, LANGUAGES.

Основна мета - алгоритмізація та програмування елементів тренажера «Граматики. Мови, що задаються граматиками» дистанційного навчального курсу «Теорія програмування».

Об'єктом розробки є процес дистанційного навчання математичним дисциплінам.

Предметом розробки є алгоритм роботи тренажеру з теми «Граматики. Мови, що задаються граматиками».

Перелік використаних методів полягає в застосуванні граматик, мов, що задаються граматиками.

Рішення будь-якої прикладної задачі з використанням електронних обчислювальних машин (ЕОМ) складається з наступних етапів:

- аналіз вимог і формальна постановка задачі;
- вибір або розробка математичної моделі;
- аналіз способів вирішення;
- логічне проектування і розробка алгоритму;
- кодування (написання програми);
- тестування і налагодження програмного забезпечення;
- впровадження, використання і супровід програмного забезпечення.

Після завершення етапу попередніх досліджень складають список вимог, що пред'являються до програмного забезпечення. Сюди входять:

- аналіз обстановки (сукупність умов, в яких передбачається експлуатувати програмну систему);
- опис виконуваних програмою системою функцій (чіткий опис того, що повинна робити система, на підставі яких вхідних даних, які дані є вихідними);
- обмеження, які повинні враховуватися в процесі проектування (терміни завершення; ресурси, наявні в наявності).

Головна мета на етапі аналізу формальної постановки задачі і складання вимог до програми полягає в пошуку і поданні таких ситуацій, які можуть привести до збою в роботі програми і у визначенні причин і способів подолання таких ситуацій.

Алгоритм робити тренажера:

Початковий крок. На панелі виводиться інформація:

- тема;
- назва дистанційного курсу;
- ПІБ автора;
- поточний рік;
- кнопка для переходу до тестування.

Якщо натиснути кнопку, то відображається перший крок.

1 крок. На панелі виводиться питання та варіанти відповіді.

При неправильно обраних варіантах вказується помилка. При правильному – перехід далі на крок.

Кінцевий крок. На панелі виводиться інформація:

- повідомлення про завершення тестування;
- перелік кроків;
- чи допущено помилку на кожному кроці;
- кнопка для повторного тестування.

Якщо натиснути кнопку, то відображається перший крок.

Застосування сучасних комп'ютерних і телекомунікаційних технологій в навчальному процесі не тільки створює умови для більш ефективної самостійної роботи студентів, сприяє індивідуалізації процесу підготовки фахівців, а і суттєво змінює форми і зміст комунікацій між викладачем і студентом. За допомогою комп'ютерних технологій, незважаючи на незмінні тенденції до зменшення аудиторних годин, прямий і зворотній зв'язок «викладач-студент» стає більш інтенсивним і активним.

Література

1. Палюх Б.В. Электронное обучение в инженерном образовании / Б.В. Палюх, А. В. Твардовский, В.К. Иванов, – 2012.– Качество образования, 10, с.34–37.
2. Інтегроване середовище розробки [Електронний ресурс] / Матеріал з Вікіпедії — вільної енциклопедії. – Режим доступу: https://uk.wikipedia.org/wiki/Інтегроване_середовище_розробки.
3. Бабій М.С. Теорія програмування: Навчальний посібник [Електронний ресурс] / М.С. Бабій, О.П. Чекалов.– Суми: Вид-во СумДУ, 2009. – 181 с.
4. Нікітченко М.С. Теоретичні основи програмування: Навчальний посібник [Електронний ресурс] / М.С. Нікітченко. – Київ: КНУ ім. Т.Г. Шевченка, 2009. – 200 с. – Режим доступу: <http://tp.unicyb.kiev.ua/doc/TOP.pdf>.

СТВОРЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ З ТЕМИ «АРИФМЕТИЧНІ ОПЕРАТОРИ Assembler»

*Пильник С. О., студент гр. КН б інт-21, спеціальності
«Комп'ютерні науки»*

*Вищий навчальний заклад Укоопспілки «Полтавський
університет економіки і торгівлі»
uu8010118@gmail.com*

*В доповіді розглядається алгоритм та створений по ньому
тренажер на тему «Арифметичні оператори Assembler».*

***Pylnyk S.O.** Creation of software for the topic «Arithmetic
operators of Assembler».*

*In the report consider the algorithm and the simulator created on
it on the topic «Arithmetic operators of Assembler».*

Ключові слова: ТРЕНАЖЕР, АРИФМЕТИЧНІ ОПЕРАТОРИ,
ASSEMBLER.

Keywords: SIMULATOR, ARITHMETIK OPERATORS,
ASSEMBLER.

В тезах викладено постановку задачі і алгоритм роботи тренажера. Головною задачею є створення програмного забезпечення у вигляді тренажера з теми «Арифметичні оператори Assembler».

Дане програмне забезпечення буде створено у середовищі розробки MS Visual Studio, з використанням мови програмування C#.

Головною задачею програмного забезпечення у вигляді тренажеру буде навчання студентів темі «Арифметичні оператори Assembler».

Після проведення пошуку в мережі інтернет, а саме на електронному ресурсі dSPACE.puet.edu.ua програм-тренажерів з

теми « Арифметичні оператори мови Assembler » було знайдено лише інформацію пов'язану з розробкою сайтів, але не програму у вигляді тренажеру для навчання конкретній темі. Саме тому вважається доцільним розробка тренажеру з теми «Арифметичні оператори мови Assembler».

Алгоритм роботи тренажера

Крок 1. На початку роботи з програмою-тренажером користувач має лише одну кнопку, для переходу в головне меню, та його інформують про те на яку тему йому надають інформацію.

Крок 2. В головному меню користувач ознайомлюється з короткою інформацією про тему яка буде тут представлена, та буде дві кнопки: перша для переходу до вибору практичного заняття (див. Крок 6), друга кнопка переносе користувача до вибору теоретичної теми для опрацювання.

Крок 3. Користувач повинен обрати тему натиснув відповідну кнопку.

Крок 4. Обрав відповідну тему користувачу буде надана відповідна до теми інформація. Тут буде дві кнопки: перша для повернення на минулу сторінку, друга для переходу до наступної сторінки.

Крок 5. На останній сторінці теми, у користувача буде дві кнопки: перша повертає до минулої сторінки, друга дає можливість відразу перейти до меню вибору теми.

Крок 6. Користувач отримує тестове завдання з декількома варіантами відповідей, після вибору правильного варіанта відповіді, вона помічається зеленим кольором та буде можливість перейти до наступного питання, але в разі не правильної відповіді, вона буде помічена червоним кольором.

Крок 7. Після проходження останнього завдання у користувача з'явиться кнопка для виходу до головного меню.

На протязі всієї роботи з програмою-тренажером користувач може натиснути на кнопку виходу, яка закінчує роботу програми.

Література

1. Арифметичні та логічні операції мови асемблер мікропроцесора Intel8086. Рік - 2013 - режим доступу:
https://knowledge.allbest.ru/programming/2c0a65625a3bd68b4c53a88421216d36_0.html

**СТВОРЕННЯ ЕЛЕМЕНТІВ ТРЕНАЖЕРА
ДИСТАНЦІЙНОГО НАВЧАЛЬНОГО КУРСУ
«МАТЕМАТИЧНИЙ АНАЛІЗ» З ТЕМИ «ОБЧИСЛЕННЯ
ГРАНИЦЬ ФУНКЦІЇ ОДНІЄЇ ЗМІННОЇ»**

***Рак А. О.**, бакалавр спеціальності «Комп'ютерні науки»
Вищий навчальний заклад Укоопспілки «Полтавський
університет економіки і торгівлі»
rak.andrii.o@gmail.com*

Розглянуто реалізацію елементів тренажера дистанційного навчального курсу «Математичний аналіз» з теми «Обчислення границь функції однієї змінної»

Rak A. O. The implementation of the elements of the simulator of the distance learning course "Mathematical Analysis" with the topics "Calculation of the boundaries of the function of one variable"

Ключові слова: ТРЕНАЖЕР, МАТЕМАТИЧНИЙ АНАЛІЗ, ФУНКЦІЯ ОДНІЄЇ ЗМІННОЇ, ОБЧИСЛЕННЯ ГРАНИЦЬ ФУНКЦІЇ.

Keywords: SIMULATOR, MATHEMATICAL ANALYSIS, FUNCTION OF ONE VARIABLE, CALCULATION OF FUNCTION LIMITS.

Одним із сучасних засобів реалізації процесу навчання на основі інформаційно-комунікаційних технологій є дистанційна освіта. Дистанційні форми і методи навчання сприяють індивідуалізації процесу професійної підготовки майбутніх фахівців, збільшують обсяг самостійної роботи, формують інформаційну культуру, наштовхують на використання інноваційних засобів знаходження та використання інформації.

Основу навчального процесу при використанні інформаційно-освітнього середовища складає цілеспрямована і

контрольована інтенсивна самостійна робота здобувача вищої освіти, котрий може навчатися в зручному для себе місці, за індивідуальним розкладом, маючи при собі комплект спеціальних засобів навчання і погоджену можливість зв'язку з керівником навчального курсу.

Використання інформаційно-освітнього середовища у ВНЗ забезпечує інформаційну насиченість та гнучкість методів навчання з використанням інформаційних технологій.

Тренажер з теми «обчислення границь функції однієї змінної» поділено на два блоки: теоретичний і практичний. Вибір здійснюється на першому вікні після запуску програми. Також після проходження одного з блоків можна перейти до іншого або розпочати знову.

Розпочавши теоретичний блок, на кожному кроці виводиться завдання та варіанти відповіді. Якщо вибрано неправильний варіант, то вказується відповідне повідомлення.

Розпочавши практичний блок, на кожному кроці виводиться завдання та поле для розв'язку. Якщо вказано неправильний розв'язок, то вказується пояснення.

Для створення тренажеру було використано мову програмування Java-код, яка володіє всіма перевагами сучасної мови програмування:

- Простий;
- Інтерпретований;
- Переносний;
- Надійний;
- Безпечний;
- Об'єктно-орієнтований;
- Високопродуктивний;
- Багатопотоковий;
- Активно розвивається.

В доповіді викладено результати програмування навчального тренажера з теми «Обчислення границь функції однієї змінної» дистанційного навчального курсу «Математичний аналіз».

Література

1. Дистанційний курс «Математичний аналіз частина 1» // Головний центр дистанційного навчання вищого навчального закладу УКООПСПЛКИ «Полтавський університет економіки і торгівлі»». – Режим доступу: <http://www2.el.puet.edu.ua/zo/course/view.php?id=1407>
2. Шкіль М.І. Вища математика: Підручник: У 3 кн.: Кн. II. Диференціальне та інтегральне числення функції однієї змінної. ряди / Шкіль М.І., Колесник Т.В., Котлова В.М.. - К.: Либідь. 1994, 352 с.
3. Кадомський К.К. Java. Теорія і практика: навчальний посібник для студентів природничих спеціальностей університетів [Текст] / К.К. Кадомський, П.К. Ніколюк / – Вінниця, Донну, 2019. – 197 с.

ТРЕНАЖЕР
«ПОБУДОВА БЛОК-СХЕМ АЛГОРИТМІВ ЦИКЛІЧНОЇ
СТРУКТУРИ НА ПРИКЛАДІ ЦИКЛУ REPEAT ... UNTIL»

А. В. Рижков, бакалавр спеціальності «Комп'ютерні науки»

Є. М. Ємець, канд. фіз.-мат. наук, проф.

О. О. Ємець, канд. фіз.-мат. наук, доц.

Полтавський університет економіки і торгівлі

Подається алгоритм тренажеру.

Ryzhkov A. V., Yemets` E. M., Yemets` O. O. Simulator «Construction of flowcharts of loop structure algorithms on the example of the repeat ... until loop». The algorithm of the simulator is given.

Ключові слова: АЛГОРИТМ, БЛОК-СХЕМА, ТРЕНАЖЕР, ЦИКЛ.

Keywords: ALGORITHM, FLOWCHART, SIMULATOR, LOOP.

Умова. Є алгоритм, записаний мовою *Object Pascal*, в якому на екран виводиться значення функції $y = e^{\sin x} \cos x$ на відріжку $[0, \pi]$ з кроком 0,1. Створити блок-схему.

```
var   x, y:real;  
const pi=3.14;  
begin  
    x:=0;  
    repeat  
        y:=exp(sin(x))*cos(x);  
        writeln('x=', x:4:2, ' y=', y:4:2);  
        x:=x+0.1;  
    until (x>pi);  
    readln;  
end.
```


На кожному кроці користувач обирає вірну відповідь із запропонованих. Якщо обрано вірний варіант (далі у алгоритмі він виділений жирним шрифтом), то з'являється підтверджуюче повідомлення і відбувається перехід на наступний крок. При помилці з'являється пояснення. Користувач повинен виправити помилку.

Крок № 1. Видимий рис. 1. У першому символі є список опцій для вибору: 1) Кінець; 2) Зупинка; **3) Початок**; 4) var.

Пояснення помилки: «Даний блок символізує старт алгоритму. Тому вірний варіант – це «Початок».

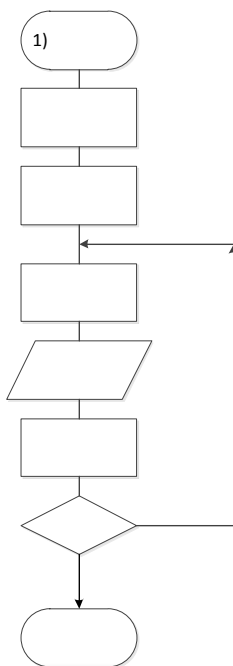


Рис. 1 – Крок 1

Крок № 2. Видимий рис. 2. У другому символі є список: 1) var; 2) var x, y:real; 3) const; **4) pi=3.14;**

Пояснення помилки: «Даний блок символізує маніпуляцію з даними. Тому вірний варіант – це « $\pi=3.14$ ».

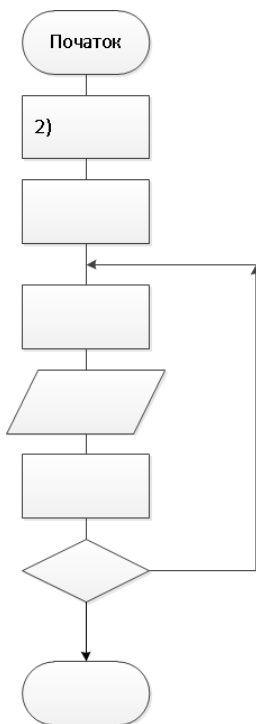


Рис. 2 – Крок 2

Крок № 3. Видимий рис. 3. У третьому символі є список:
1) $x=0$; 2) var x, y:real; 3) repeat; 4) var.

Пояснення помилки: «Даний блок символізує маніпуляцію з даними. Тому вірний варіант – це « $x=0$ ».

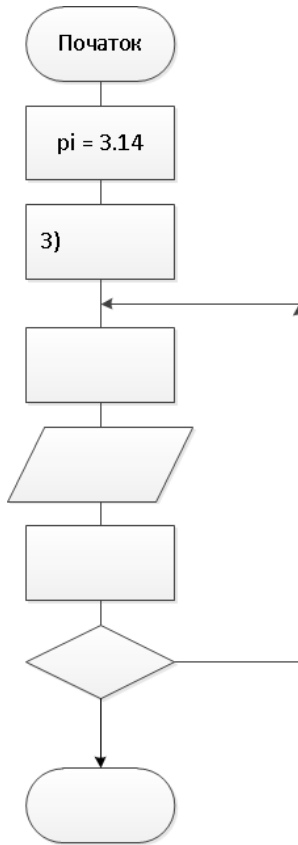


Рис. 3 – Крок 3

Крок № 4. Видимий рис. 4. У четвертому символі є список:
 1) repeat; 2) $y = e^{\sin x} \cos x$; 3) writeln('x=', x:4:2, ' y=', y:4:2);
 4) $x > \pi$.

Пояснення помилки: «Даний блок символізує маніпуляцію з даними. Тому вірний варіант – це $y = e^{\sin x} \cos x$ ».

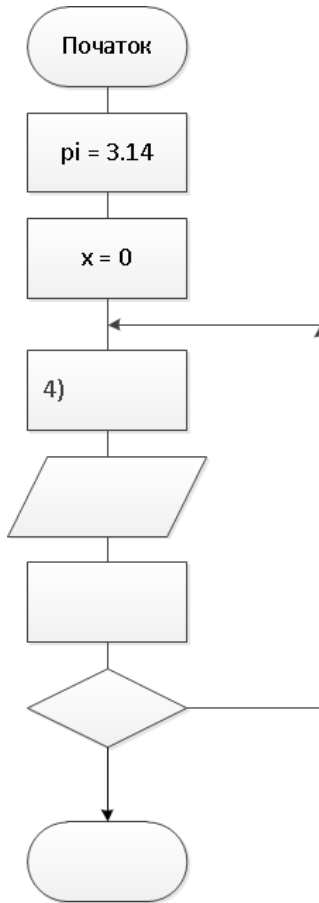


Рис. 4 – Крок 4

Крок № 5. Видимий рис. 5. У п'ятому символі є список:
1) Вивід x та y ; 2) repeat; 3) $x := x + 0.1$; 4) $x > pi$.

Пояснення помилки: «Даний блок символізує вивід даних. Тому вірний варіант – це «Вивести x та y ».

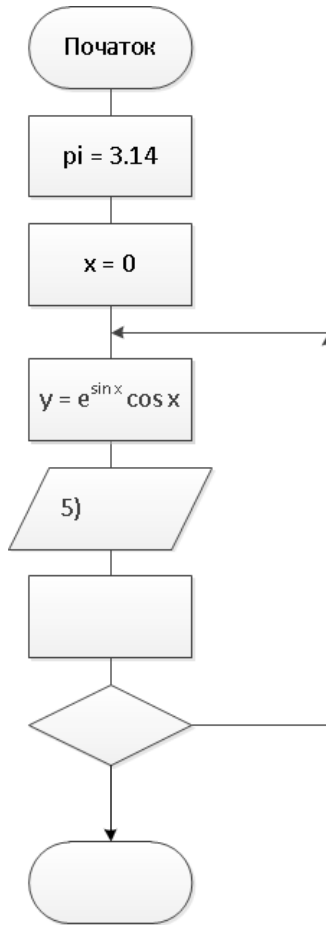


Рис. 5 – Крок 5

Крок № 6. Видимий рис. 6. У шостому символі є список:
 1) Вивід x та y ; 2) $x > \pi$; 3) $x = x + 0.1$; 4) until ($x > \pi$);

Пояснення помилки: «Даний блок символізує маніпуляцію з даними. Тому вірний варіант – це « $x = x + 0.1$ ».

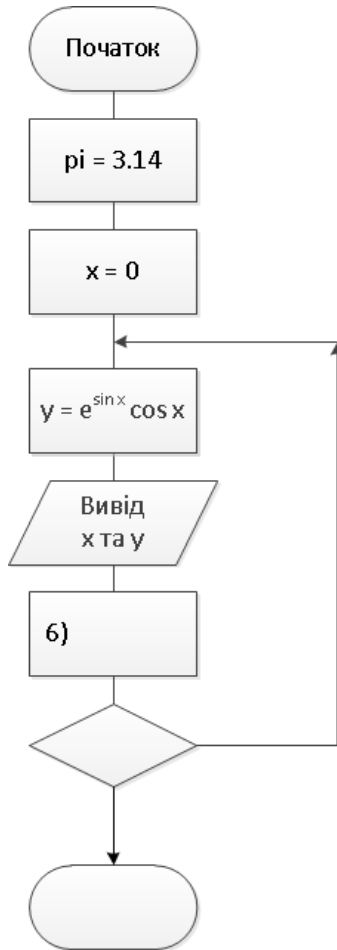


Рис. 6 – Крок 6

Крок № 7. Видимий рис. 7. У цьому символі є список:
 1) repeat; 2) readln; 3) until; **4) $x > \pi$?**

Пояснення помилки: «Даний блок символізує перевірку умови. Тому вірний варіант – це « $x > \pi$?».

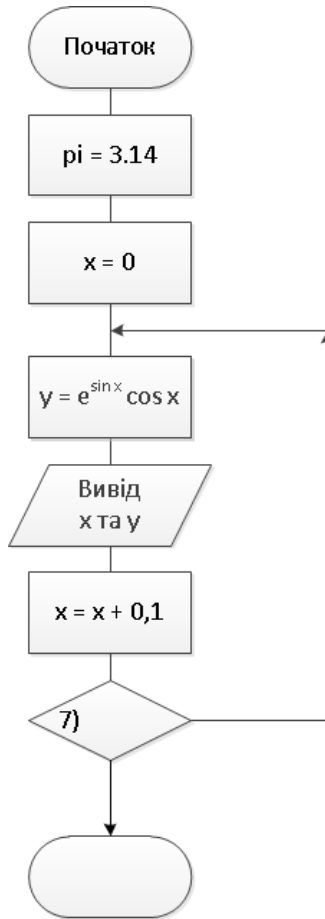


Рис. 7 – Крок 7

Крок № 8. Видимий рис. 8. На лінії (біля цифри 8) є список:
1) так; 2) *ні*.

Пояснення помилки: «Цикл repeat ... until виконується тоді, коли умова циклу не виконується. Отже, вірний варіант – «ні».

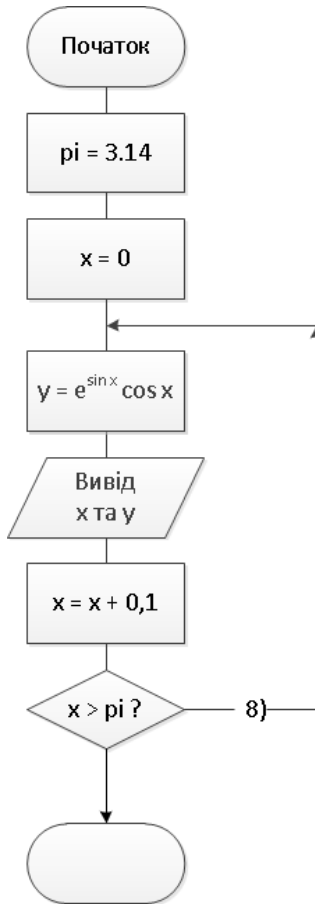


Рис. 8 – Крок 8

Крок № 9. Видимий рис. 9. На лінії (біля цифри 9) є список:
 1) так; 2) ні.

Пояснення помилки: «Цикл repeat ... until завершує роботу тоді, коли умова циклу виконується. Отже, вірний варіант – «так».

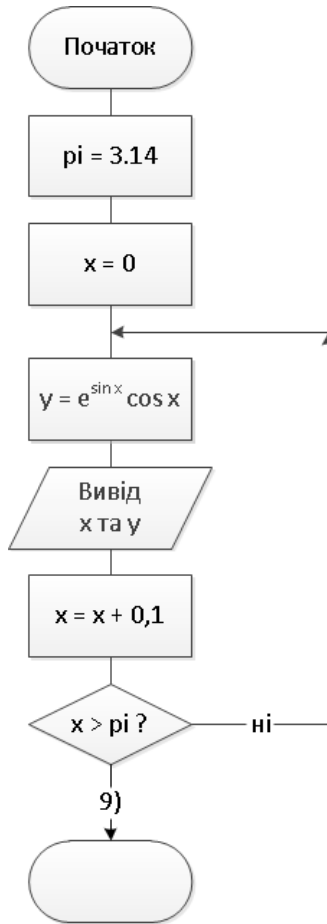


Рис. 9 – Крок 9

Крок № 10. Видимий рис. 10. У символі (біля цифри 10) список: 1) Початок; 2) **Кінець**; 3) readln; 4) repeat.

Пояснення помилки: «Даний блок символізує завершення алгоритму. Тому вірний варіант – це «Кінець».

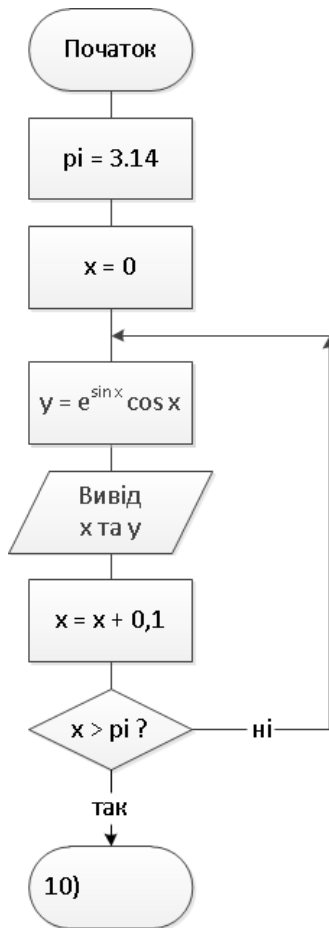


Рис. 10 – Крок 10

Крок № 11. Таким чином, блок-схему алгоритму приймає вигляд (рис. 11):

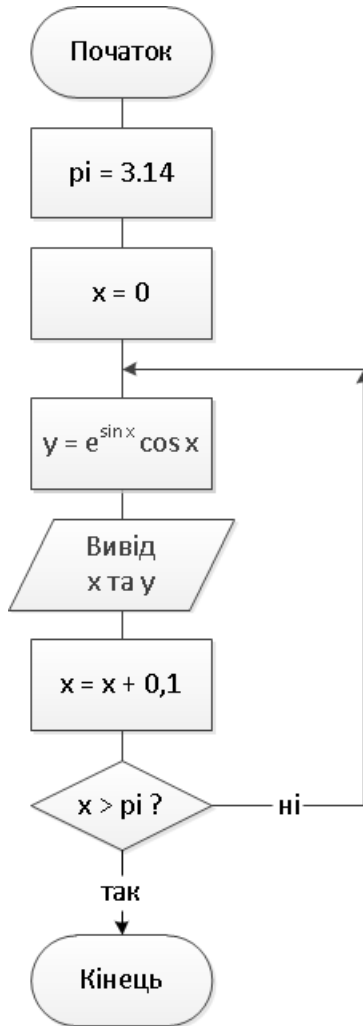


Рис. 11 – Крок 11

Література

1. Ємець О. О. Дистанційний курс ПУЕТ «Інформатика. Частина 1» для студентів спеціальності «Комп'ютерні науки» / О. О. Ємець. – [Електронний ресурс].

2. Бибка Б. М. Тренажер «Побудова блок-схем алгоритмів циклічної структури на прикладі циклу while» / Б. М. Бибка, О. О. Ємець // Комп'ютерні науки і прикладна математика (КНіПМ-2020): матеріали наук.-практ. семінару. Випуск 6. / За ред. Ємця О. О. – Полтава: Кафедра ММСІ ПУЕТ, 2020. – Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/10039>.
3. Ємець О. О. Про розробку тренажерів для дистанційних курсів кафедру ММСІ ПУЕТ / О. О. Ємець // Інформатика та системні науки (ІСН-2015): матеріали VI Всеукр. наук.-практ. конф. за міжн. участю (м. Полтава, 19-21 березня 2015 р.) / за ред. Ємця О. О. – Полтава: ПУЕТ, 2015. – С. 152-161. – Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/2488>.
4. Ємець О. О. Про розробку тренажерів для дистанційних курсів навчальних дисциплін спеціальності Комп'ютерні науки в ПУЕТ / О. О. Ємець, Є. М. Ємець, О. О. Ємець // Економіка сьогодні: проблеми моделювання та управління: матеріали X Міжн. наук.-практ. інтернет-конф. (м. Полтава, 19-20 листопада 2020 р.). – Полтава: ПУЕТ, 2021. – С. 365-370.
5. Мордасова І. В. Тренажер з теми «Побудова блок-схем алгоритмів розгалуженої структури» дистанційного навчального курсу «Інформатика» та розробка його програмного забезпечення / І. В. Мордасова, О. О. Ємець // Комп'ютерні науки і прикладна математика (КНіПМ-2019): матеріали наук.-практ. семінару. Випуск 3. / За ред. Ємця О. О. – Полтава: Кафедра ММСІ ПУЕТ, 2019. – С. 35-37. – Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/7037>.
6. Сузанська А. О. Тренажер «Побудова блок-схем алгоритмів розгалуженої структури» / А. О. Сузанська, Є. М. Ємець, О. О. Ємець // Комп'ютерні науки і прикладна математика (КНіПМ-2020): матеріали наук.-практ. семінару. Випуск 5. / За ред. Ємця О. О. – Полтава: Кафедра ММСІ ПУЕТ, 2020. – С. 56-62. – Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/8906>.

**РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ
ТРЕНАЖЕРУ З ТЕМИ «МАГАЗИННІ АВТОМАТИ»
ДИСТАНЦІЙНОГО НАВЧАЛЬНОГО КУРСУ «ТЕОРІЯ
ПРОГРАМУВАННЯ»**

***В.В. Лукіна, студентка гр. КН б інт-21, спеціальності
«Комп'ютерні науки»***

*Вищий навчальний заклад Укоопспілки «Полтавський
університет економіки і торгівлі»*

О. О. Черненко, к. ф.-м. н., доцент

Вищий навчальний заклад

Укоопспілки «Полтавський

університет економіки і торгівлі»

Популярність дистанційної освіти в останні роки різко зросла. Ця форма навчання є найбільш гнучкою та доступною для багатьох бажаючих отримати знання. Багато сказано на користь дистанційної освіти, і тому розробка тренажерів для дистанційних курсів стає з кожним днем актуальнішою [1].

Метою проекту було створення програмного забезпечення з теми «Магазинні автомати» дистанційного навчального курсу «Теорія програмування».

Плюсами розробленого програмного забезпечення є:

- Зручний та інтуїтивно зрозумілий інтерфейс
- Наявність перевірки на правильність введеної відповіді
- Можливість повторити роботу з тренажером по його складовим частинам
- Виведення результату на кожному кроці роботи
- Можливість повернутися в меню або до умови завдання в будь-який момент.

Програмне забезпечення розроблене за допомогою платформи UNITY 2018 на мові програмування C#. Для написання коду була використана Visual Studio 2017.

При розробці тренажеру акцент стояв на засвоєнні матеріалу із курсу «Теорія програмування» як практичного, так і теоретичного. Матеріал теоретичної частини викладено в вигляді тестових завдань аналогічно з практичною частиною.

На початковому екрані програми-тренажера користувач отримує інформацію з розглянутої теми тренажеру та його розробника. Також на початковому екрані існує можливість переходу до завдань теоретичного чи практичного типу за допомогою відповідних кнопок.

При натисненні на кнопку – «почати роботу» користувач переходить до теоретичної частини тренажеру, в якій розміщені тестові завдання до теми.

Після вибору правильного варіанту відповіді користувач отримує доступ до кнопки «Далі» та повідомлення про правильну відповідь, у іншому випадку користувач отримує повідомлення про помилку.

По закінченню роботи з теоретичною частиною, користувач переходить до практичної частини, де отримує умову та тестові завдання для її розв'язування.

Після вибору правильної відповіді відбувається перехід на екран з поточним прогресом у розв'язку задачі. Закінчивши роботу, користувач отримує кінцевий результат та може повторити роботу з тренажером.

Література

1. <http://www.i.ua/>Кафедра менеджменту Львівського торговельно-економічного університету навчання [Електронний ресурс] - Режим доступу до ресурсу <https://kerivnyk.info>
2. Український центр дистанційного навчання [Електронний ресурс] - Режим доступу до ресурсу www.distance-learning.com.ua
3. Математический форум Math Help Planet Обсуждение и решение задач по математике, физике, химии, экономике [Електронний ресурс] - Режим доступу до ресурсу

<http://mathhelpplanet.com/static.php?p=magazinnyye-avtomaty>

4. Дистанционное обучение: плюсы и минусы.
[Электронный ресурс] - Режим доступа до ресурсу
<https://kpi.ua/ru/805-9>

РОЗВ'ЯЗУВАННЯ ДИФЕРЕНЦІАЛЬНИХ РІВНЯНЬ ПЕРШОГО ПОРЯДКУ

Ю.О.Ткаленко, студент гр. КН б інт-21, спеціальності
«Комп'ютерні науки»

Вищий навчальний заклад Укоопспілки «Полтавський
університет економіки і торгівлі»

*В доповіді розглядається алгоритм та створений по ньому
тренажер з теми «Розв'язування диференціальних рівнянь
першого порядку».*

Tkalenko Y.O. *Creation of software on the topic "Expansion of
differential levels of the first order". The algorithm and the
simulator for the topic " Expansion of differential levels of the first
order " are considered in the report.*

Ключові слова: ТРЕНАЖЕР, ДИФЕРЕНЦІАЛЬНІ
РІВНЯННЯ, МАТЕМАТИЧНИЙ АНАЛІЗ.

Keywords: SIMULATOR, DIFFERENTIAL EQUATIONS,
MATHEMATICAL ANALYSIS.

В тезах викладено постановку задачі тренажера.

Основною задачею бакалаврської роботи являється
програмна реалізація веб-додатка з теми «Розв'язування
диференціальних рівнянь першого порядку»

Завданням даного курсового проекту є:

1. Вибір способу програмної реалізації веб-додатка;
2. Складання алгоритму роботи програмного забезпечення;
3. Складання блок-схеми алгоритму роботи програмного
забезпечення;
4. Програмна реалізація веб-додатка;
5. Перевірка валідності та тестування програмного

забезпечення;

Основними вимогами до програмного забезпечення є:

1. Зрозумілість видачі інформації;
2. Пояснення кроків;
3. Перегляд кількості вірних відповідей;
4. Можливість перегляду кроків програми через довідку;
5. Повідомлення користувача про помилку або не вибрані елементи тесту;

Створено сайт для навчання темі «Розв’язування диференціальних рівнянь першого порядку».

Література

1. difur.in.ua [Електронний ресурс]: Диференціальні рівняння
Режим доступу до сайту:
<http://difur.in.ua/studentam/lektsii/lc03/>
2. disted.edu.vn.ua [Електронний ресурс]: Диференціальні рівняння
Режим доступу до сайту:
<https://disted.edu.vn.ua/courses/learn/1283>

ЦИКЛ FOR В EXCEL VBA

*Д. В. Малишева, студентка гр. КН б інт-21, спеціальності
«Комп'ютерні науки»
Вищий навчальний заклад Укоопспілки «Полтавський
університет економіки і торгівлі»
malyshevadaria8@gmail.com*

*В доповіді розглядається алгоритм та створений по ньому
тренажер з теми «Цикл for в Excel VBA».*

***Malysheva D. V.** Creation of software on the topic "For loop in
Excel VBA". The algorithm and the simulator for the topic "For loop
in Excel VBA" are considered in the report.*

Ключові слова: ТРЕНАЖЕР, ЦИКЛ FOR, EXCEL VBA.
Keywords: SIMULATOR, FOR LOOP, EXCEL VBA.

В тезах викладено постановку задачі тренажера.

1. Основними завданнями даного курсового проекту з фаху є:
2. Програмна реалізація елементів тренажеру з теми «Цикл for в Excel VBA».
3. Розробка алгоритму роботи з програмним забезпеченням.
4. Розробка блок-схем алгоритму роботи.
5. Основними вимогами до програмного забезпечення є:
6. Надійність програмного забезпечення.
7. Зручна навігація в програмному забезпеченні.
8. Наявність практичних завдань з теорією на одному екрані.

Створено тренажер для навчання темі «Цикл for в Excel VBA».

**ТРЕНАЖЕР «ВІДНОШЕННЯ. ОБЛАСТЬ ВИЗНАЧЕННЯ,
ОБЛАСТЬ ЗНАЧЕНЬ, ГРАФ, МАТРИЦЯ
ВІДПОВІДНОСТІ, ПЕРЕРІЗ ЗА ЕЛЕМЕНТАМИ»
ДИСТАНЦІЙНОГО НАВЧАЛЬНОГО КУРСУ
«ДИСКРЕТНА МАТЕМАТИКА»**

***Борута І. В.**, студент спеціальності «Комп'ютерні науки та інформаційні технології»*

ВНЗ Укоопспілки «Полтавський університет економіки і торгівлі»
ivan.boruta@gmail.com

***Парфьонова Т. О.**, к.ф.-м.н., доцент*

ВНЗ Укоопспілки «Полтавський університет економіки і торгівлі»
tpa.poltava@gmail.com

В доповіді розглядається алгоритм та програмне забезпечення тренажера з теми «Відношення. Область визначення, область значень, граф, матриця відповідності, переріз за елементами» дистанційного курсу «Дискретна математика».

Boruta I.V., Parfonova T.O. In the report the algorithm and implementation of the simulator «Relation. Domain, range, graph, the correspondence matrix, intersection on the element» for distance learning course Discrete mathematic are considered.

Ключові слова: ДИСКРЕТНА МАТЕМАТИКА, ВІДНОШЕННЯ, ТРЕНАЖЕРИ, АЛГОРИТМ.

Keywords: DISCRETE MATHEMATIC, RELATION, SIMULATOR, ALGORITHM.

Важливу роль в навчальному процесі має застосування програм-тренажерів, які дозволяють краще засвоїти матеріал з даної теми, зокрема, отримати практичні навички розв'язування

типових прикладів. Створення таких продуктів залишається актуальним. [1]

Даний тренажер реалізує наступні задачі: знаходження області визначень та області значень відношення, представлення відношення графом та матрицею відповідності, побудова перерізу по елементу та перерізу по множині відношення. [2] Розглянемо алгоритм реалізації знаходження перерізів відношень за елементами.

Приклад. Нехай задано відношення ρ на множинах $A = \{1, 2, 3, 4\}$ та $B = \{a, b, c, d, e\}$, $\rho \subseteq A \times B$, що задане множиною впорядкованих пар:
 $\rho = \{(1, b), (2, a), (2, c), (4, e), (1, c)\}$.

Знайти всі перерізи за елементами відношення ρ .

Алгоритм.

Крок 1. На екрані з'являється перше запитання:

Вибрати одне правильне означення перерізу за елементом відношення ρ .

- Множина $A(x_i)$ таких y , що пари $(y, x_i) \in \rho$ називається перерізом за елементом x_i відношення ρ .

- Множина $A(x_i)$ пар $(x_i, y) \in \rho$ називається перерізом за елементом x_i відношення ρ .

- Множина $A(x_i)$ таких y , що пари $(x_i, y) \in \rho$ називається перерізом за елементом x_i відношення ρ .

- Множина $A(x_i)$ таких пар (x_i, y) , що $(x_i, y) \notin \rho$ називається перерізом за елементом x_i відношення ρ .

Якщо обрано правильну відповідь, то здійснюється перехід на наступний крок. Інакше – відображається повідомлення про помилку: «Відповідь не правильна. Множина $A(x_i)$ таких y , що пари $(x_i, y) \in \rho$ називається перерізом за елементом x_i відношення ρ . ». Перехід на крок 2.

Крок 2. Наступне питання:

«Знайдемо перерізи відношення ρ – $A(1)$, $A(2)$, $A(3)$ та $A(4)$. Скільки елементів у цих множинах?

Ввести відповідні значення у комірці:

$$|A(1)| = \square, |A(2)| = \square, |A(3)| = \square, |A(4)| = \square.»$$

Користувач одне за одним вводить значення у комірці. При цьому кожна наступна комірка стає активною для введення лише після того, як в попередню введено правильну відповідь.

Правильна відповідь має бути наступною:

$$1) |A(1)| = \boxed{2}, |A(2)| = \boxed{2}, |A(3)| = \boxed{0}, |A(4)| = \boxed{1}.$$

Якщо введено інші значення, то з'являються повідомлення про помилку:

1) Кількість елементів множини $A(1)$ визначається кількістю пар $(1, y) \in \rho$, тобто $(1, b)$ та $(1, c)$. Отже, $|A(1)| = 2$.

2) Кількість елементів множини $A(2)$ визначається кількістю пар $(2, y) \in \rho$, тобто $(2, a)$ та $(2, c)$. Отже, $|A(2)| = 2$.

3) Кількість елементів множини $A(3)$ визначається кількістю пар $(3, y) \in \rho$. Таких пар немає у ρ . Отже, $|A(3)| = 0$.

4) Кількість елементів множини $A(4)$ визначається кількістю пар $(4, y) \in \rho$, тобто $(4, e)$. Отже, $|A(4)| = 1$. Перехід на крок 3.

Крок 3. Завдання на екрані.

Вибрати елементи, які належать перерізу.

Переріз за елементом 1: $A(1) \bullet a \bullet b \bullet c \bullet d \bullet e \bullet \emptyset$.

Переріз за елементом 2: $A(2) \bullet a \bullet b \bullet c \bullet d \bullet e \bullet \emptyset$.

Переріз за елементом 3: $A(3) \bullet a \bullet b \bullet c \bullet d \bullet e \bullet \emptyset$.

Переріз за елементом 4: $A(4) \bullet a \bullet b \bullet c \bullet d \bullet e \bullet \emptyset$.

Правильний вибір – на екрані з'являються відповідно відповіді: $A(1) = \{b, c\}$, $A(2) = \{a, c\}$, $A(3) = \emptyset$, $A(4) = \{e\}$.

Якщо вибір помилковий, то з'являються повідомлення про помилку відповідно для кожного перерізу:

1) Маємо дві пари $(1, b)$ та $(1, c)$, що належать відношенню ρ . Тоді за означенням $A(1) = \{b, c\}$.

2) Маємо дві пари $(2, a)$ та $(2, c)$, що належать відношенню ρ . Тоді за означенням $A(2) = \{a, c\}$.

3) Не маємо жодної пари $(3, y) \in \rho$, тому $A(3) = \emptyset$.

4) Маємо одну пару $(4, e)$, що належить відношенню ρ . Тоді за означенням $A(4) = \{e\}$.

Алгоритм прикладу завершено.

Для створення програми було обрано мову програмування C++ та середовище Borland Builder.

Література

1. Ємець О. О. Про розробку тренажерів для дистанційних курсів кафедрою ММСІ ПУЕТ / О. О. Ємець // Інформатика та системні науки (ІСН-2015): матеріали VI Всеукр. наук.-практ. конф. за міжн. участю (м. Полтава, 19-21 березня 2015 р.) / за ред. Ємця О. О. – Полтава: ПУЕТ, 2015. – С. 152-161. – Режим доступу: <http://dSPACE.puet.edu.ua/handle/123456789/2488>.

2. Ємець О.О. Дискретна математика: Навчальний посібник. Вид. 2-ге, допов. / О.О.Ємець, Т.О.Парфьонова. – Полтава: РВВ ПУСКУ, 2009. – 287 с. – Режим доступу: <http://dSPACE.uccu.org.ua/handle/123456789/552>

**ALGORITHMIZATION AND DEVELOPMENT OF
SIMULATOR ON THE TOPIC “CYCLES IN JAVA” OF THE
DISCIPLINE “PROGRAMMING**

Lartey W. H., Bachelor, Student majoring in 122 «Computer Sciences»

Poltava University of Economics and Trade.

larteysurviver@gmail.com

Oleksiichuk Yu., Candidate of Sciences.

Poltava University of Economics and Trade

olexijchuk@gmail.com

The algorithm of the simulator on the topic “Cycles in Java” is discussed in the thesis.

Keywords: SIMULATOR, DISTANT LEARNING, CYCLES IN JAVA.

In the thesis, considers the subject of the development the software application for the simulator on the topic «Cycles in Java» of the distant course «Programming»

Purpose of the thesis is the development of the algorithm of the training for solving cycles in java problems by the means of using a simulator, the software implementation of program testing.

Methods that have been used to solve the problem is, Algorithm theory methods are used in the thesis work. The program was developed using Java language in the environment of Netbeans IDE 8.2.

The software implementation put into practice that, it gives students a platform to enhance their learning skills and capabilities through the simulator.

The testing revealed that, the simulator is easy to use and has a good readability for students to work with.

The thesis developed a simulator algorithm.

The simulator can be implemented as a sequence of questions. The student must answer each question. If a student gives an incorrect answer, he will receive a hint and will answer again.

Objective: Develop a java program that contains questions and answers with hints to users in a graphical user interface.

Problem Data.

Example 1. What is the result of executing the following code?

```
for(int i=10;i>=5;i--){  
    System.out.print(2*i+" ");  
}
```

A) 20 18 16 14 12 10 (*)

B) 20 18 16 14 12

C) 10 9 8 7 6 5

D) 10 9 8 7 6

E) 9 8 7 6 5

Hints:

If student choose B)

The hint: "wrong answer"

If student choose C)

The hint: "wrong answer"

If student choose D)

The hint: "wrong answer"

Descriptive Algorithm

Algorithm.

Step 0. Start the program.

Step 1. Display the first question, possible list of answers.

Step 2. User then makes a choice.

Step 3. If user selects a wrong choice, a pop-up message appears as a hint, move to step 2.

Step 4. User clicks on the next button.

Step 5. If list of questions is empty, move to step 7.

Step 6. Display the next question, possible list of answers.

Step 7. End program.

The report gives a solution problem that, in all, I successfully use an algorithm and chose a suitable method and a platform to create questions and answers for students to acquire and enhance their learning capabilities through the simulator.

Reference

1. Create a Simple Window Using JFrame – Access mode:
<https://www.thoughtco.com/create-a-simple-window-using-jframe-2034069>
2. If Else Statement in Java –Access mode:
<https://www.topperskills.com/tutorials/java/java-if-else-syntaxes-example.html>
3. Do-While Loop- Access mode:
https://funnelgarden.com/java-for-loop/#Do-While_Loop
4. How to create JButton in Java- Access mode:
<https://javapointers.com/java/java-se/the-jbutton/>
5. JRadioButton - Part 1-Access mode:
<https://www.macs.hw.ac.uk/cs/java-swing-guidebook/?name=JRadioButton&page=1>

DEVELOPMENT OF A SIMULATOR OF THE DISTANCE LEARNING COURSE

N. S. Chagonda, Bachelor's, Student majoring in 122 «Computer Sciences» Poltava University of Economics and Trade.
stanchagonda@gmail.com

In the article are discussed the creation of an algorithm and a simulator that are able to assist a student or students on the subject "algebra and geometry" on the topic "a plane in space".

Keywords: SIMULATOR, ALGEBRA, GEOMETRY, PROGRAM, PLANE, SPACE, JAVA, FXML.

The main objective of the project work is to try and increase student's access to information and assist them in mastering particular topics using an application that can potentially be downloaded from the internet.

The user is presented with a series of questions, in their language of choice, in which they can only advance to the next question if they have chosen a correct answer. In the case that the user has selected a wrong answer, a prompt message appears on the screen providing assistance to the user in the form of a hint. The program ends when all the questions are completed.

The simulator was created first by writing the algorithm and then drew a series of flowcharts. After that Java FXML was used to code and create the final simulator.

The simulator can be further developed and improved by adding other topics and also maybe by connecting the program to a database in order to allow easy updates of content.

Reference

1. Bloch Joshua. Effective Java. 3rd Edition. Released: 27 December 2017. Editor: Addison-Wesley Professional. ISBN-13: 978-0134685991.

2. Brown G. Introducing FXML A Markup Language for JavaFX / Greg Brown. – 2011. Access mode:<http://fxexperience.com/wp-content/uploads/2011/08/Introducing-FXML.pdf>.

**ELEMENTS FOR THE SIMULATOR ON THE TOPIC
"ALGORITHM FOR CONDUCTING THE LEFT
FACTORIZATION OF GRAMMAR" OF THE DISTANCE
LEARNING COURSE "PROGRAMMING THEORY"**

A. Darboe. *Bachelor of Computer Science
Ukoospilka Higher Educational Institution "Poltava University of
Economics and Trade"*
amzdabz@gmail.com

*Elements of the simulator on the topic "Algorithm for conducting
the left factorization of grammar" of the distance learning course
"Programming Theory" is proposed.*

Keywords: SIMULATOR, LEFT FACTORIZATION,
PROGRAMMING THEORY.

The main purpose is to create elements of the simulator software on the topic "Algorithm for conducting the left factorization of grammar" of the distance learning course "Programming Theory".

The object of the course project is the process of distance learning in mathematical and computer science disciplines.

The subject of the course project is a software product that implements a simulator on the topic of "Algorithm for conducting the left factorization of grammar".

The simulator is solve through Java programming language
The main tasks of the work:

- describe the problem statement;
- describe an overview of the purposes of distance learning and learning management systems;
- give the basic concepts of the topic for use in the simulator;
- develop an algorithm for the simulator and make a flow chart;
- describe the programming language and technologies used in developing the program;

- describe the process of implementation of the main stages of creating a simulator;
- Describe the instruction required by the user.

Creating simulators for distance learning opens up a new way for students of distance form of education. The advantage of simulators is that they can be used both for student training and for self-study.

The initial window of the simulator should display the following information about the simulator:

- topic;
- surname, initials of the student;
- Academic group of the student.

The following issues should be covered in the simulator according to the topic:

1. Recursive descent.
2. Algorithm of deleting of left recursion.
3. Left factorization.

Consider the operation of the simulator based on testing.

Step 1. The user is given the task: “Recursion is one of the fundamental programming methods. While program creating, it is often necessary to reduce the initial tasks to simpler ones.

The method of reducing a task to itself, but with modified initial data, is called:

(answer options are provided)

- recursion,
- recursive descent,
- left factorization”

If the answer is correct, i.e. the first option is selected, then there is a transition to the next step. If not, an error message is displayed.

Step 2. The user is given the task: “Recursive descent is:

(answer options are provided)

- a procedure, based on mutually not recursive equivalents, each of which implements one of the products of grammar,
- an algorithm of parsing, based on recursive, each of which implements one of the products of grammar,

- an algorithm of parsing, based on mutually recursive procedures (or not recursive equivalents), each of which implements one of the products of grammar”

If the answer is correct i.e. the third option is selected then there is a transition to the next step. If not, an error message is displayed.

An algorithm simulator for the topic "Algorithm for conducting the left factorization of grammar" of the distance learning course "Programming Theory" was developed over all.

Literature

1. Keleş Kaya Mümine. A Review of Distance Learning and Learning Management Systems / Mümine Kaya Keleş, Selma Ayşe Özel.– London: IntechOpen Limited, DOI: 10.5772/65222. – Access Mode:
<https://www.intechopen.com/books/virtual-learning/a-review-of-distance-learning-and-learning-management-systems>
2. Бабій М.С. Теорія програмування: Навчальний посібник [Електронний ресурс] / М.С. Бабій, О.П. Чекалов.– Суми: Вид-во СумДУ, 2009. – 181 с.
3. Нікітченко М.С. Теоретичні основи програмування: Навчальний посібник [Електронний ресурс] / М.С. Нікітченко. – Київ: КНУ ім. Т.Г. Шевченка, 2009. – 200 с. – Режим доступу: <http://tpr.unicyb.kiev.ua/doc/TOP.pdf>.
4. Zykov S.V. Modern programing languages. Part I. Functional approach to programing / S.V. Zykov. – М.: MIFI, 2003. – 230 p.

**SOFTWARE ELEMENTS FOR THE SIMULATOR ON THE
TOPIC "PREDICTIVE PARSING: SCHEME, PRINCIPLE OF
OPERATION, APPLICATION" OF THE DISTANCE
LEARNING COURSE "PROGRAMMING THEORY"**

Muhammad Adnan, Bachelor of Computer Science
Ukoopspilka Higher Educational Institution "Poltava University of
Economics and Trade"
adnanpervaiz1994@hotmail.com

Software elements of the simulator on the topic "Predictive parsing: scheme, principle of operation, application" of the distance learning course "Programming Theory" is proposed.

Keywords: SIMULATOR, PREDICTIVE PARSING, SCHEME, PRINCIPLE OF OPERATION.

The main purpose is to develop elements of the simulator software on the topic "Predictive parsing: scheme, principle of operation, application" of the distance learning course "Programming Theory".

The object of the course project is the process of distance learning in mathematical disciplines.

The subject of the course project is the algorithm of the simulator on the topic "Predictive parsing: scheme, principle of operation, application".

The list of methods used is the use of predictive parsing.

Non-recursive predictive analyzer can be constructed using explicit use of the stack instead implicit in the recursive calls. The key problem of predictive analysis is to identify products that need to be applied to non-terminal as well. A parse table can be used to find products.

Formulation of the problem:

1. Given an analyzer, use it to analyze the input stream. Namely: to learn to determine the products that need to be used at a certain step.

2. Learn to read and be able to compile a spreadsheet.
3. Create an algorithm for using a predictive analyzer.
4. Develop an algorithm simulator.

The simulator program provides:

- sequential display of tasks;
- control over the user's actions to solve the proposed task;
- instant reaction to wrong actions;
- demonstration of the correct solution of the problem;
- output of the final message about

The list of educational applications, platforms and resources below aim to help parents, teachers, schools and school administrators facilitate student learning and provide social care and interaction during periods of school closure. Most of the solutions curated are free and many cater to multiple languages.

The algorithm of the simulator outlines the main steps of constructing predicative parsers for further software implementation.

The list of the main tasks and reference information on them is given:

- operation of the analyzer program;
- algorithm 1. Non-recursive predictive analysis;
- lgorithm 2. Construction of the set FIRST for grammar symbols;
- algorithm 3. Construction of FOLLOW (X) for all X-nonterminals of grammar;
- algorithm 4. Construction of predictive analysis tables.

After passing the completion message is displayed, the start page opens.

If the answer is incorrect, the output of reference information is implemented.

Creating simulators for distance learning - this opens up a new way for us to study for students of distance (distance) form of education. The advantage of simulators is that they can be used both for student training and for self-study.

The main results of the work:

1. Created theoretical material for the simulator;
2. Created tests for the simulator;

3. Developed an algorithm for the simulator on the topic "Predictive parsing: scheme, principle of operation, application" of the distance learning course "Programming Theory".

Literature

1. Toolkit: Tools for Creating Software Simulations [digital resource] / Joe Ganci. – Available from: <https://learningsolutionsmag.com/articles/toolkit-tools-for-creating-software-simulations>
2. Distance learning solutions [digital resource] / WWW.UNESCO.ORG. – Available from: <https://en.unesco.org/covid19/educationresponse/solutions>
3. Бабій М.С. Теорія програмування: Навчальний посібник [Електронний ресурс] / М.С. Бабій, О.П. Чекалов.– Суми: Вид-во СумДУ, 2009. – 181 с.
4. Нікітченко М.С. Теоретичні основи програмування: Навчальний посібник [Електронний ресурс] / М.С. Нікітченко. – Київ: КНУ ім. Т.Г. Шевченка, 2009. – 200 с. – Режим доступу: <http://ttp.unicyb.kiev.ua/doc/TOP.pdf>.

Алфавітний покажчик авторів

Бибка Б. М.	40
Борута І. В.	91
Бурко А. О.	51
Деркач Р. А.	13
Дудник Д. А.	54
Закіров Р.Р.	60
Іжевський Д.О.	63
Ємець Є.М.	72
Ємець О.О.	5, 9,13
Ємець Ол-ра О.	40,72
Клименко М.М.	25
Купченко О. В.	35
Лукіна В.В.	85
Малишева Д. В.	90
Олексійчук Ю.Ф., Oleksiichuk Yu. F.	21, 95
Олефіренко В. В.	37
Парфьонова Т.О.	35, 54, 91
Пилипченко В. С.	5
Пильник С. О.	66
Рак А. О.	69
Рижков А. В.	72
Стельнік А.І.	16
Ткаленко Ю.О.	88
Фесенко Д. І.	21
Цехмейстер В. О.	9
Черненко О.О.	16, 85
Чілікіна Т.В.	25
Darboe A.	100
Lartey W. H.	95
Muhammad A.	103
Chagonda N. S.	98