

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Факультет комп'ютерних наук та кібернетики
Кафедра теорії та технології програмування

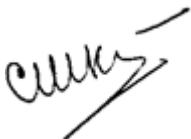
Кваліфікаційна робота
на здобуття ступеня бакалавра
за спеціальністю 122 Комп'ютерні науки
на тему:

**РОЗРОБКА ЕЛЕКТРОННОГО ПОСІБНИКА З
КУРСУ «ТЕОРІЯ АЛГОРИТМІВ»**

Виконав студент 4-го курсу
Ілля РИЧЕНКО



(підпис)

Науковий керівник:
доктор фіз.-мат. наук, професор
Степан ШКІЛЬНЯК


(підпис)

Засвідчую, що в цій роботі немає
запозичень з праць інших авторів
без відповідних посилань

Студент


(підпис)

Роботу розглянуто й допущено до
захисту на засіданні кафедри теорії
та технології програмування

«01» червня 2022 р.

протокол No 10

Завідувач кафедри
Микола НІКІТЧЕНКО

(підпис)

РЕФЕРАТ

Обсяг роботи 33 сторінки, 7 ілюстрацій, 9 джерел посилань.

ВЕБ-ПРОЄКТ, ВІЗУАЛІЗАЦІЯ, ІНТРЕФЕЙС, КОРИСТУВАЛЬНИЦЬКИЙ ФУНКЦІОНАЛ, НАВІГАЦІЯ, ТЕОРІЯ АЛГОРИТМІВ

Об'єкт роботи: електронні навчальні матеріали.

Мета роботи: розробка веб-проєкту у вигляді електронного посібника з курсу «Теорія алгоритмів», що дає можливість студентам ознайомитись з матеріалом курсу за допомогою зручного інтерфейсу та зрозумілого меню керування для переходів між темами посібника та візуалізація формальних моделей алгоритмів (МНР програма та машина Тюрінга).

Інструменти розроблення: редактор коду Microsoft Visual Studio Code – інтегроване середовище розробки (IDE), що підтримує мову JavaScript, TypeScript та бібліотеку користувальницького інтерфейсу React; HTML – стандартна мова розмітки для створення веб-сторінок та веб-додатків; CSS – спеціальна мова стилів для опису зовнішнього вигляду сторінок, написаних мовами розмітки даних; Sass – скриптова метамова, яка компілюється в звичайні CSS-стили, система контролю версіями GitHub.

Методи розроблення: побудова веб-проєкту навчального посібника, який містить основні відомості з обраного курсу, з використанням сучасних технологій та інструментів веб-розроблення.

Результати роботи: розроблено веб-проєкт, який дозволяє користувачам ознайомлюватись з темами курсу «Теорія алгоритмів» за допомогою зручного та зрозумілого користувальницького інтерфейсу.

Сфера застосування: робота може знайти використання серед студентів в освоєнні матеріалів з курсу «Теорія алгоритмів».

Значимість роботи: забезпечення переходу від письмових до сучасних електронних способів вивчення та освоєння курсу «Теорія алгоритмів». Зручність у використанні, інтуїтивно зрозумілий інтерфейс та оптимізація витраченого часу на пошук необхідного матеріал

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ	4
ВСТУП	5
РОЗДІЛ 1. ОГЛЯД ІСНУЮЧИХ СИСТЕМ.....	7
РОЗДІЛ 2. ДИСТАНЦІЙНА ОСВІТА ТА МУЛЬТИМЕДІЙНІ СИСТЕМИ..	9
2.1 Поняття дистанційної освіти	9
2.2 Інформаційні технології в освітньому процесі.....	11
2.3 Поняття електронного навчального посібника та принципи його побудови	14
РОЗДІЛ 3. ІНСТРУМЕНТИ РОЗРОБКИ.....	17
3.1 Visual Studio Code	17
3.2 TypeScript.....	18
3.3 React	20
3.4 Sass	22
РОЗДІЛ 4. РЕАЛІЗАЦІЯ ВЕБ-ПРОЄКТУ.....	23
4.1 Створення програмного інтерфейсу	23
4.2 Моделювання формальних моделей алгоритмів	27
4.3 Вимоги до ПЗ.....	29
ВИСНОВКИ	30
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	32
ДОДАТКИ	33

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ

IDE – Integrated Design Environment, інтегроване середовище розробки.

HTML – Hyper Text Markup Language, мова гіпертекстової розмітки.

CSS – Cascading Style Sheets, каскадні таблиці стилів.

SASS – Syntactically Awesome Style Sheets, синтаксично вражаючі таблиці стилів.

МНР - машина з натуральнозначимими регістрами.

JSX – JavaScript XML.

DOM – Document Object Model, об'єктна модель документа.

ОС – операційна система.

ПЗ – програмне забезпечення.

ВСТУП

Оцінка сучасного стану об'єкта розробки. В наші дні бурхливий розвиток інформаційних технологій сприяє проникненню та використанню їх в багатьох галузях людської діяльності. Водночас ще далеко не у всіх сферах життєдіяльності людини впроваджується їх використання. На сьогоднішній день у викладанні різних курсів переважає вживання друкованих підручників або у вигляді звичайних електронних форматів. Проте, це не завжди буває зручно та інколи потребує додаткового часу для знаходження необхідного навчального матеріалу. У випадку, коли треба швидко знайти необхідну інформацію з тієї чи іншої теми, нам доводиться власноруч шукати її, опрацьовуючи чималу кількість джерел. У такому разі ми витрачаємо більше часу на пошук матеріалів, ніж на їх освоєння.

Актуальність роботи та підстави для її виконання. Одним із недоліків аналогічних додатків є те, що в переважній більшості вони містять інформацію іноземною мовою. Також ці додатки не завжди мають зручний і зрозумілий для користувача інтерфейс. Авжеж, ці фактори можуть ускладнювати та уповільнювати процес використання додатку, адже користувачу необхідний час для отримання навичок для його користуванням. Тому для того, щоб не витрачати час на освоєння якихось нових форм опанування навчального матеріалу або на їх переклад, людина може повернутись до форм, більш звичних та зрозумілих для неї. Для розробників під час створення додатку необхідно, перш за все, брати до уваги зручність користування цим додатком з погляду користувача. Водночас зауважимо, що зручність – доволі суб'єктивний критерій. Наприклад, деякі люди вважають основним функціональну складову додатку, а інші – візуальний інтерфейс.

Мета й завдання роботи. Розробка веб-проєкту у вигляді

електронного підручника з курсу «Теорія алгоритмів» зі зручним інтерфейсом для опанування студентами основних тем курсу.

Для досягнення даної мети було поставлено такі завдання:

- Аналіз існуючих проєктів.
- Ознайомитись із сучасними інструментами та технологіями веб-розробки, які застосовуються в тому числі на комерційному рівні.
- Підготувати необхідні матеріали, для наповнення ними додатку.
- Розробити відповідний веб-проєкт.

Об'єкт, методи й засоби розроблення. Об'єктом розроблення є програмна реалізація електронного підручника з курсу «Теорія алгоритмів» у вигляді веб-застосунку, що дозволить студентам в зручнішому форматі опанувати матеріал. Для інструмента розроблення було обрано: редактор коду Microsoft Visual Studio Code – інтегроване середовище розробки (IDE), що підтримує мову JavaScript, TypeScript, використовувалась бібліотека користувальницького інтерфейсу – React та спеціальні файлові розширення, такі як, JSX/TSX, для комбінування JavaScript/TypeScript коду та HTML – стандартна мова розмітки для створення веб-додатків, CSS – спеціальна мова, що використовується для опису зовнішнього вигляду сторінок.

Можливі сфери застосування. Додаток може знайти використання серед студентів для зручного ознайомлення з матеріалами відповідного навчального курсу.

РОЗДІЛ 1. ОГЛЯД ІСНУЮЧИХ НА РИНКУ СИСТЕМ

На сьогоднішній день існує декілька інструментів, які можна використовувати в цілях розробки електронних навчальних матеріалів. Проте, ці інструменти не завжди є простими в розробленні та зручними для користувачів, що може перешкодити як розробникам, так і користувачам даних програмних продуктів. До найпопулярніших відносяться, зокрема:

- **LaTeX.** Це система підготовки документів, яка включає в себе набір інструментів, які з текстових файлів, записаних з використанням спеціальної мови розмітки, формують готові документи. В більшості його використовують для створення наукових документів, написання книг та інших форм публікацій. Завдяки великій кількості бібліотек, що знаходяться у відкритому доступі, LaTeX зручно використовувати для роботи зі схемами, математичними виразами, таблицями та ін. Також стилізація шрифтів та інших елементів в документі узгоджується для всього файлу одразу, редагування вихідного коду можливе у будь-якому текстовому редакторі. Водночас даний інструмент має низку недоліків:
 - необхідність знати мову спеціальної розмітки для створення документів;
 - відсутні можливості веб-додатків, такі як: швидка навігація між темами, додавання відео, додавання різноманітних стилів.
- **TurboSite.** Безкоштовна програма для створення сайтів та електронних підручників з метою їх розміщення на безкоштовних хостингах. Інструмент доволі простий у користуванні, не потребує знання мов програмування та розмітки тексту. Також застосунок містить багато готових шаблонів, що дозволяє прискорити процес розробки. Попри ряд переваг, TurboSite має деякі недоліки:

- можливе некоректне відображення в різних середовищах;
 - іноземна мова інтерфейсу та обмежений вибір сервісів для хостингу готового застосунку;
 - неможлива персоналізована стилізація та обмежений функціонал веб-додатку.
- eBooksWriter LITE. Проста в експлуатації програма з великою кількістю функцій. Електронний підручник створений з використанням цього застосунку має вигляд *.exe файлу, який може містити аудіо та відеоматеріали. Головним недоліком цього продукту є те, що безкоштовна версія дозволяє створювати електронні підручники розміром до 1 мегабайту.

РОЗДІЛ 2. ДИСТАНЦІЙНА ОСВІТА ТА МУЛЬТИМЕДІЙНІ СИСТЕМИ

2.1 Поняття дистанційної освіти

Розвиток інформаційних технологій актуалізував проблему розвитку та вдосконалення системи освіти. Суть цього вдосконалення найбільше виразилась в понятті дистанційної освіти.

Дистанційна освіта (ДО) визначається як специфічна форма організації навчального процесу із використанням спеціальних технологій та засобів комунікації між викладачем та здобувачем освіти. Характерною рисою дистанційного навчання є те, що в переважній більшості, студенти та викладачі віддалені між собою у просторі або в часі. Дистанційне навчання є перспективним способом отримання освіти ізольованим сільським учням, які навчаються із специфічними вимогами або особами, нездатними досягти поставленої мети іншим способом.

Незмірно зросла роль дистанційної освіти в останні два роки. Це найперше пов'язано з пандемією COVID-19 та воєнним станом в Україні, введеним через агресію росії.

Важливою особливістю дистанційного навчання є те, що їх методи взаємодії з освітніми матеріалами, потребують мінімальної участі викладача та студента. Для розвитку цих методів використовується мультимедійний підхід, в якому за допомогою різних засобів створюються відео, аудіо та графічні матеріали. До таких методів відносяться[1] насамперед:

- електронні журнали(для контролю успішності студентів);
- інтерактивні бази даних;
- електронні підручники.

Засобами, які забезпечують у дистанційній освіті використання всіх зручних форм взаємодії, є інформаційні телекомунікаційні технології. Так,

Інтернет, дає змогу здобувачам освіти самостійно вивчати та знайомитись з різними джерелами інформації. До таких засобів можна віднести: відеоконференції, електронна пошта, віртуальний клас, чат.

Відеоконференція – послуга, яка надає можливість користувачам обмінюватись аудіо- та відеоінформацією в режимі реального часу, що є надзвичайно важливим при комунікації між викладачами та здобувачами освіти на відстані.

Електронна пошта – інтернет-сервіс, що надає змогу обмінюватись повідомленнями як в текстовому форматі, так і в графічному, аудіо та відео форматах. Дана форма надає змогу студенту листуватись з викладачем для обговорення та обміну інформацією. Електронна пошта є одним з найшвидших способів обміну повідомленнями.

Віртуальний клас – спільнота учнів, які взаємодіють між собою при виконанні навчальних завдань за допомогою комп'ютерних мереж. Такий засіб дозволяє викладачу поширювати завдання одразу для всіх студентів, перевіряти їх та зручно оцінювати.

Чат – спілкування між користувачами в режимі реального часу, з використанням різноманітних додатків для обміну повідомленнями. Спілкування в чатах може відбуватись в текстовому, аудіо та відеорежимі.

Основними завданнями дистанційного навчання є (див.[2]):

- розширення доступу різних категорій учнів до якісного навчання;
- індивідуалізація процесу навчання згідно до потреб і особливостей здобувачів освіти;
- підвищення ефективності навчального процесу шляхом впровадження новітніх технологій;
- створення додаткових можливостей для спілкування педагогічних працівників з студентами та студентів між собою в рамках активного творчого засвоєння програми навчання;
- контроль за якістю освіти.

Розвиток дистанційної освіти буде вдосконалюватися та покращуватись

разом з розвитком інформаційних технологій та вдосконалення методів самого дистанційного навчання.

Таким чином, завдання, які вирішує дистанційна освіта та швидкий розвиток інструментів та технологій, покликаних вдосконалити їх використання, показують необхідність розширення та їх широкого впровадження. Дистанційна освіта дозволяє студентам знайомитись з новими для них формами навчання, покращує їх самоорганізацію та ефективність самостійної роботи.

2.2 Інформаційні технології в освітньому процесі

На сьогоднішній день важко уявити діяльність будь-якого навчального закладу, чи то школи, чи закладів вищої освіти, без використання інформаційних технологій. Інформаційні технології (ІТ) – це набір методів та засобів, що використовуються для збору, зберігання, обробки та поширення інформації.

В наші дні бурхливий розвиток інформаційних технологій сприяє проникненню та використанню їх в багатьох галузях людської діяльності. Звичайно, інформаційні технології не могли обійти таку важливу сферу людської діяльності, як освіта. Впровадження комп'ютерів в освіту дозволило підвищити якість навчання, ефективніше налагодити комунікацію між студентом та викладачем. Інформатизація навчального процесу дає змогу викладачеві краще подати навчальний матеріал, зробивши його цікавим, та підвищує рівень сприйняття інформації студентами. Але для широкого використання ІТ у закладах освіти необхідна деяка модернізація, перехід від традиційних засобів обміну інформацією до використання комп'ютерних технологій.

Інформатизація освіти, як складова частина глобальної інформатизації, є системою методів, процесів і програмно-технічних засобів, інтегрованих з метою збору, обробки, збереження, поширення і використання інформації в інтересах її споживачів.

Використання ІТ в освітньому процесі має низку переваг (див.[3]):

- забезпечення комунікації викладачів та студентів та студентів між собою поза межами навчального закладу;
- зменшення часу, необхідного на пошук потрібних навчальних матеріалів;
- використання спеціальних систем для організації навчального процесу, для підвищення рівня пізнання інформації студентами;
- розвиток систем дистанційної освіти різного рівня;
- підвищення контролю за якістю надання освіти.

Основними завданнями інформатизації освіти є:

- підвищення якості підготовки спеціалістів за рахунок використання інформаційних технологій у навчальному процесі;
- адаптація навчання відповідно до індивідуальних особливостей здобувачів освіти;
- забезпечення неперервності в навчанні.

Таким чином, засоби нових інформаційних технологій дозволяють не тільки надавати студентам якомога більше готових, строго відібраних та відповідним чином організованих знань, умінь, навичок, але й розвивати інтелектуальні творчі здібності, уміння самостійно здобувати нові знання, працювати з різними джерелами інформації.

Мультимедійні засоби навчання займають важливе місце у розвитку інформаційного суспільства та у впровадженні інформаційних технологій в навчальний процес. Мультимедійні засоби навчання – це комплекс програмних засобів, що дозволяють користувачеві в інтерактивній формі сприймати інформацію, таких як: графіку, гіпертексти, аудіо, анімацію, відео. Мультимедійні системи надають користувачеві персонального комп'ютера такі види інформації: текст, зображення, анімаційні картини, аудіо коментарі, цифрове відео.

Існують різноманітні способи використання мультимедійних технологій в навчальному процесі, серед яких:

- використання електронних підручників, посібників;
- візуалізація різноманітних процесів і явищ;
- забезпечення дистанційної форми навчання;
- проведення інтерактивних освітніх відеоконференцій;
- побудова систем контролю й перевірки знань і навичок студентів, створення онлайн тестів;
- створення сайтів навчальних закладів;
- створення презентацій навчального матеріалу.

Узагальнюючи факти, перелічені вище, можемо зробити висновок, що використання засобів мультимедіа в освітньому процесі сприяє (див.[4]):

- підвищенню сприйняття та засвоєння студентами навчального матеріалу;
- реалізації соціальної мети, а саме – інформатизації суспільства;
- індивідуалізації процесу навчання;
- розвитку особистості студента;
- розвитку самонавчання.

Загалом, впровадження комп'ютерів в освітньому процесі привело до покращення якості інформаційних освітніх технологій, які змогли підвищити якість навчання, зробили зручнішим та швидшим процес взаємодії студентів з викладачами.

2.3 Поняття електронного навчального посібника та принципи його побудови

Однією з складових як у дистанційній освіті, так і у використанні інформаційних технологій у навчальному процесі, є електронні навчальні посібники та підручники.

Електронний навчальний посібник – це програмний комплекс з навчальними матеріалами, що відповідає навчальній програмі з конкретного предмета. Електронний посібник призначений для допомоги в оволодінні навчальною дисципліною. Матеріал у ньому викладений у зручній для вивчення формі, а також може містити у собі різноманітні графічні матеріали та тести для кращого засвоєння інформації.

Під час створення електронного посібника потрібно спочатку підготувати необхідну літературу, сформулювати зміст, форматування тексту на модулі відповідно до розробленого змісту та створити необхідні мультимедійні матеріали (аудіо-, відео- або графічні матеріали).

Окрім різного формату, електронний посібник має ряд суттєвих переваг від друкованого посібника:

- можливість застосування мультимедіа;
- можливість створення зручної та простої навігації;
- структуризація навчального матеріалу;
- індивідуальний підхід до здобувачів освіти;
- можливість адаптації навчального матеріалу відповідно до рівня знань студентів;
- можливість контролю рівня знань;
- інтерактивність, що дозволяє встановити зворотній зв'язок між студентом та джерелом інформації.

У побудові електронних навчальних посібників можна виокремити наступні принципи (див.[5], ст. 25-26):

1. Науковість. Зміст посібника має містити знання в тій формі, яка відповідає науковій моделі у відповідній галузі.
2. Доступність. Представлення навчального матеріалу засобами інформаційних технологій повинно відбуватися у вигляді моделей, які є адекватними для сприйняття студентами.
3. Наочність. Навчальна діяльність повинна будуватись з залученням як аудіо так і відео матеріалів, які наочно можуть продемонструвати навчальні моделі.
4. Свідоме ставлення до навчання. Активізація самонавчання студента – щодо знаходження необхідної інформації.
5. Індивідуальність навчання. Має забезпечуватись індивідуальний підхід до здобувачів освіти, з урахуванням їх можливостей та особливих потреб.

Електронні навчальні посібники можуть бути використані для наступних цілей:

- організація дистанційного навчання;
- самонавчання студентів;
- контроль засвоєних знань;
- теоретичне навчання.

Отже, структура, наповнення, склад електронного навчального посібника можуть відрізнятися від друкованого, так як він мусить містити у собі мультимедійні технології – гіперпосилання, інтерактивні бази даних, закладки тощо.

Електронні навчальні продукти – явище досить нове й чекає на широке впровадження в навчальному процесі.

РОЗДІЛ 3. ІНСТРУМЕНТИ РОЗРОБКИ

3.1 Visual Studio Code

Visual Studio Code [6] – це редактор коду, розроблений компанією Microsoft та доступний для операційних систем Windows, macOS, Linux. Редактор позиціонується як «легкий» інструмент для розробки веб- та хмарних додатків. Цей інструмент є одним з найбільш популярних для веб-розроблення. Для роботи було використано VS Code версії 1.67, в якій були усунені деякі несуттєві недоліки пов'язані з розширеннями.

Visual Studio Code має ряд суттєвих переваг у порівнянні з іншими інтегрованими середовищами розробки, зокрема:

- підтримка розроблення консольних застосунків, веб-додатків та мобільних застосунків;
- підтримка багатьох популярних мов програмування, таких як: JavaScript, TypeScript, C++, C#, Java, Python та інші.
- вбудовані інструменти для роботи з системою контролю версіями Git;
- одночасна робота з декількома проєктами, можливість розділення інтерфейсу для паралельної роботи з декількома файлами;
- бібліотека розширень, які допомагають полегшити роботу розробників та пришвидшують процес розробки;
- навігаційне меню, що допомагає користувачам швидко переміщуватись між файлами;
- наявність глобального пошуку, для знаходження потрібної інформації серед файлів проєкту;
- широка бібліотека шаблонів та готових елементів з можливістю додавання власних елементів;
- невеликий розмір даного редактору коду дозволяє користуватись ним на малопотужних комп'ютерах;
- зручна документація з постійною підтримкою, в якій можна

ознайомитись з інструкцією для початку роботи в редакторі, та поширеними питаннями з якими можуть зіткнутися початківці.

3.2 TypeScript

Відомо, що для написання веб-додатків використовується мова програмування JavaScript, але ця мова має ряд суттєвих недоліків, один з яких динамічна типізація. Через це розробники мусять або використовувати JavaScript з усіма його недоліками, або використовувати інші мови, які компілюються в JavaScript, наприклад, Dart, CoffeeScript, ClojureScript чи TypeScript. Саме остання була обрана для розроблення додатку.

TypeScript [7] – це строго типізована мова програмування, що компілюється в JavaScript. І в JavaScript і в TypeScript є підтримка об'єктно-орієнтованого програмування (ООП): об'єкти, класи, успадкування. Але TypeScript використовує більше можливостей ООП, наприклад:

- інтерфейси;
- модифікатори доступу (public, private та protected);
- абстрактні класи, та інші.

TypeScript набрав широкої популярності серед розробників через ряд переваг в порівнянні з іншими мовами, серед яких:

- статистична типізація, що допомагає уникнути помилок, які можуть виникнути при використанні мов з динамічною типізацією;
- компіляція - TypeScript компілюється і перевіряє помилки;
- реалізація концепцій, які властиві об'єктно-орієнтованим мовам;
- можливість написання більш складних та комплексних програм, які простіше підтримувати та розвивати;
- підтримка модулів;
- складніше допустити помилку в процесі розроблення;
- відкритий код проєкту.

Типи даних, які використовуються в TypeScript:

- вбудовані типи, до яких відносяться string, number, boolean;

- об'єднання - таким чином ми можемо присвоїти одній змінній одразу декілька типів, які можуть бути присвоєні їй;
- `void` – тип який використовується для вихідного значення функцій;
- `never` – тип, що не дозволяє помістити в себе значення;
- `any` – тип який «вмикає» динамічну типізацію, таким чином, змінна типу `any` може містити в собі будь-яке значення;
- `unknown` – невідомий тип, в якому не має властивостей та методів;
- `Generics` – особливий тип даних, що дозволяє створювати змінні в типах.

Але у використанні TypeScript є деякі недоліки, зокрема:

- впровадження даної мови програмування в проєкті збільшує час на розроблення продукту;
- через збільшення часу розроблення зростає і вартість самої розробки;
- не усі сторонні бібліотеки та фреймворки JavaScript оптимізовані до TypeScript, і через це розробникам необхідно власноруч описувати деякі сигнатури, для цього потрібно витратити додатковий час;
- поріг входження є вищим, на відміну від JavaScript.

Враховуючи всі перелічені факти, можна зробити висновок, що TypeScript є чудовим вибором мови програмування для веб-застосунків, особливо для досвідчених розробників, які мають досвід у веб-розробці. Зручний синтаксис та широкий вибір різноманітних типів даних дозволяє створювати комплексні застосунки, які в подальшому простіше підтримувати.

3.3 React

Останні роки, JavaScript є однією з найбільш популярних мов програмування та користується великим попитом серед розробників. Враховуючи такі тенденції, корисно використовувати в розробці JavaScript фреймворки – це бібліотеки програмування, в яких є попередньо написаний код для використання в стандартних функціях та задачах програмування. Звичайно, програмування можливе і без використання сторонніх бібліотек та фреймворків, але правильно обране середовище може значно полегшити роботу за рахунок написання меншої кількості програмного коду. Серед популярних фреймворків можна виокремити наступні: Vue, Angular та React. Саме останній, був обраний для використання при розробці проєкту.

React [8] – це декларативна JavaScript-бібліотека для розробки інтерфейсу користувача. Була розроблена компанією Facebook та активно використовується такими компаніями як Instagram, WhatsApp, Airbnb, Netflix ті іншими. За допомогою React створюються веб-застосунки, які змінюють відображення елементів без перезавантаження сторінки.

Широкої популярності серед розробників, React набув завдяки ряду суттєвих переваг в порівнянні з іншими JavaScript бібліотеками, а саме:

- віртуальний DOM – це концепція програмування, в якій «віртуальне» представлення інтерфейсу користувача зберігається в пам'яті, та синхронізується з реальним DOM. Якщо відбуваються які-небудь зміни, спочатку змінюється віртуальний DOM, потім використовуючи спеціальний алгоритм порівняння зміни переносяться у справжній DOM. Це допомагає покращити продуктивність додатків, не витрачаючи час на зайві та непотрібні перезавантаження;
- використання спеціальних розширень – JSX/TSX. Ці файлові розширення допомагають суттєво пришвидшити розробку за рахунок можливості писати як JavaScript код так і HTML;
- компонентний підхід. Повторне використання вже готових візуальних

компонентів є однією з головних концепцій використання React;

- можливість розробки не лише веб-додатків, а також застосунків для мобільних пристроїв, використовуючи ReactNative;
- декларативний підхід у створенні користувацького інтерфейсу;
- спільнота розробників. Популярність даної бібліотеки викликала створення великої спільноти розробників, які постійно працюють над її вдосконаленням та постійним розвитком.

Разом з React рекомендується використовувати бібліотеку react router, які доповнюють одна одну. React router – це стандартна бібліотека маршрутизації в React. Основним компонентом в React router є Link – спеціальний тег, який дозволяє переходити між сторінками в додатку.

Таким чином, компонентно-орієнтований підхід та використання таких концепцій, як віртуальний DOM, дозволяє швидко створювати високопродуктивні застосунки. Факт, що бібліотеку розробляють і вдосконалюють висококваліфіковані розробники, дає підстави сподіватися, що тенденції до популяризації і широкого використання даної бібліотеки серед розробників будуть зберігатись і надалі.

3.4 Sass

Невід'ємною складовою веб-розробки є стилізація. Стили для веб-сторінок створюються за допомогою CSS.

CSS – це спеціальна мова для візуального відображення вмісту HTML-документу. CSS дає змогу змінювати або налаштовувати колір, розмір та стиль тексту, колір фону, розташування елементів на сторінці, розмір елементів та багато іншого. Для зручного використання CSS веб-розробники користуються такими інструментами, як Sass.

Sass – це CSS-препроцесор, що надає додатковий набір функцій для більш зручного та швидкого написання стилів. За допомогою Sass можна використовувати змінні, для зменшення кількості коду, використання математичних функцій, вкладення стилів. Ці можливості роблять використання Sass для стилізації набагато зручнішим в порівнянні з CSS та суттєво прискорюють процес розробки.

РОЗДІЛ 4. РЕАЛІЗАЦІЯ ВЕБ-ПРОЄКТУ

Проаналізувавши схожі за функціоналом та призначенням програмні продукти та технології для їх розробки, було створено веб-додаток у вигляді електронного посібника з курсу «Теорія алгоритмів».

Реалізація даного веб-додатку відбувалась у декілька етапів:

1. Створення програмного інтерфейсу.
2. Моделювання формальних моделей алгоритмів.

4.1 Створення програмного інтерфейсу

Під час створення електронного навчального посібника було враховано основні недоліки існуючих додатків. Тому було прийнято рішення розробити простий та зручний користувацький інтерфейс, який не буде містити зайвої інформації.

Основними інструментами розробки було обрано мову програмування TypeScript та бібліотеку React. Використовувались також додаткові інструменти, такі як HTML, CSS для розмітки сторінки та стилізації відповідно. Для зручності процесу розробки було використано CSS-препроцесор - Sass, який за рахунок використання додаткових функціональних можливостей значно пришвидшує та спрощує процес розробки. Також були застосовані сторонні бібліотеки, такі як React router та react icons. Перша дозволяє створювати маршрутизацію (навігацію) між сторінками додатку. Друга, в свою чергу, надає доступ до великої кількості зображень, для використання їх при стилізації додатку.

В результаті було створено меню для навігації, що дає змогу користувачеві переходити між сторінками електронного посібника. Елементи меню відображають основні теми курсу посібника (рисунок 4.1).

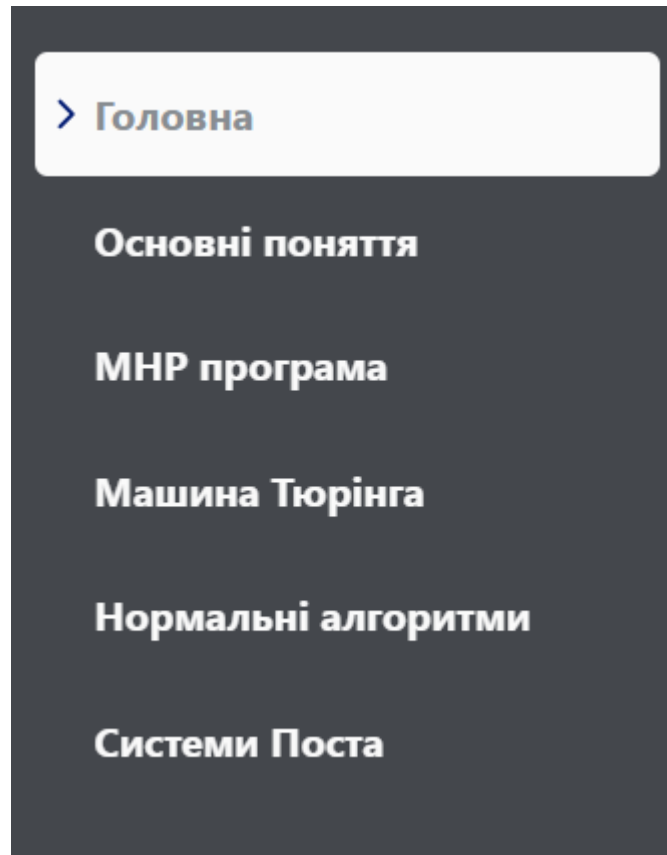


Рисунок 4.1 – Навігаційне меню веб-додатку

При переході на будь-яку сторінку через меню навігації, обраний розділ виділяється кольором та додається спеціальне зображення.

На головній сторінці додатку відображається вступний текст додатку та джерело, з якого були взяті матеріали для наповнення електронного посібника, при натисканні на дане джерело відбувається завантаження джерела у вигляді архіву (рисунок 4.2).



Теорія алгоритмів

Вступ до теорії алгоритмів

У техніку термін алгоритм прийшов разом з кібернетикою. Поняття алгоритму допомогло, наприклад, точно визначити, що означає ефективно задати послідовність керуючих сигналів. Застосування ЕОМ послужило стимулом розвитку теорії алгоритмів і вивчення алгоритмічних моделей, до самостійного вивчення алгоритмів з метою їх порівняння з роботи даного продукту (числу дій, витраті пам'яті), а також їх оптимізації.

Теорія алгоритмів зробила істотний вплив на розвиток ЕОМ і практику програмування. У теорії алгоритмів були передбачити основні концепції, закладені в апаратуру і мови програмування ЕОМ. Так, мікропрограмування будується на ідеях машин Тюрінга, структурне програмування запозичило свої конструкції з теорії рекурсивних функцій, мови символічного обробки інформації (РЕФАЛ, ПРОЛОГ) беруть початок від нормальних алгоритмів Маркова і систем Поста.

Теорія алгоритмів – наука про загальні властивості алгоритмів. Вона визначає та досліджує формальні моделі алгоритмів і алгоритмічно обчислюваних функцій.

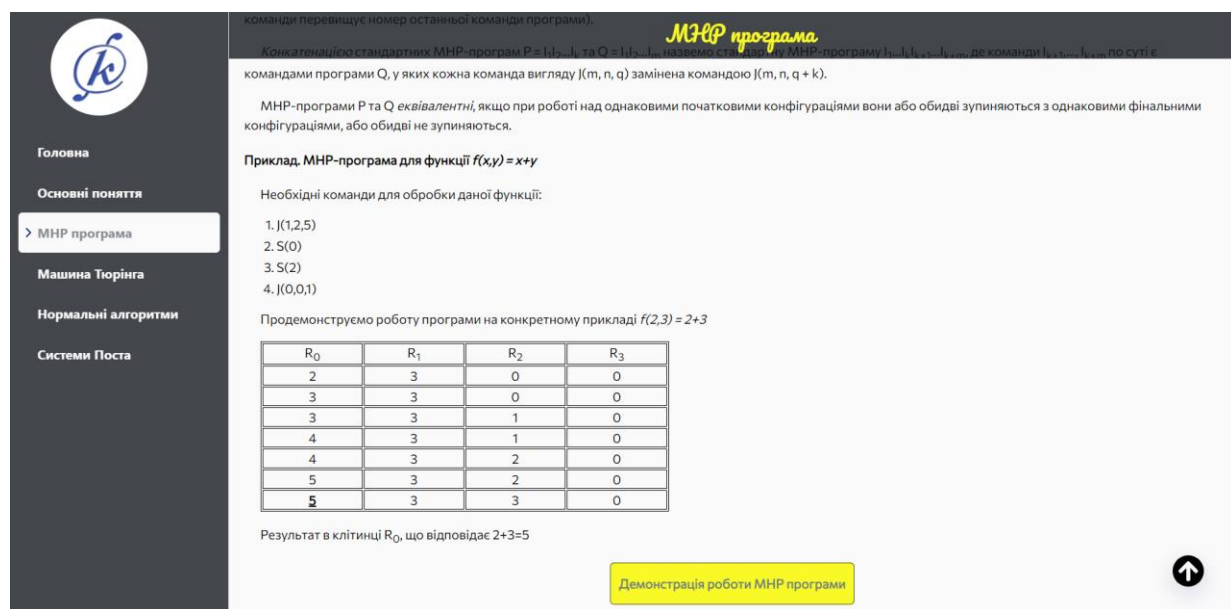
Загальна Теорія алгоритмів займається проблемою ефективної обчислюваності. Розроблено кілька формальних визначень алгоритму, в яких ефективність і кінцівку обчислень може бути визначена кількісно – числом елементарних кроків і обсягом необхідної пам'яті.

На сучасному рівні теорія алгоритмів тісно пов'язана з теоретичною математикою, що надає їй апарат для дослідження розв'язності проблем та з прикладною математикою, що вивчає моделювання явищ реального світу. У теорії алгоритмів розроблені та використовуються різні алгоритмічні системи, кожна з яких призначена для побудови алгоритмів розв'язання задач певного класу. Це такі системи, як рекурсивні функції, нормальні алгоритми Маркова, машини Поста та Тюрінга, абстрактні автомати, формальні граматики та інші. У сучасній програмній інженерії алгоритми, як методи вирішення завдань, займають провідне місце в порівнянні з традиційною математикою. Причому не важливо, існують чи ні чисте алгоритмічне рішення в абстрактних моделях алгоритмів. Якщо рішення задачі необхідно, широко використовується евристика, а "доказом" працездатності алгоритму є успішне його тестування.

Джерело матеріалів для електронного посібника: М.С. Нікітченко, О.С. Шкільняк, С.С. Шкільняк, **Теорія алгоритмів: навчальний посібник** – Київ: Видавничо-поліграфічний центр "Київський університет", 2015. –241 с.

Рисунок 4.2 – Головна сторінка

Всі сторінки навчального посібника наповнені основними теоретичними відомостями з обраних тем, а також наведені деякі практичні приклади. В розділах, які відповідають формальним моделям алгоритмів (МНР програми та машина Тюрінга), окрім теоретичних відомостей на прикладів, наведені графічні матеріали з демонстрацією їх роботи (рисунок 4.3-4.4).



команди перевищує номер останньої команди програми).

МНР програма

До кожного стандартного МНР-програми $P = \{I_1, I_2, I_3, \dots, I_n\}$ та $Q = \{I_1, I_2, I_3, \dots, I_m\}$ назвемо ступенями МНР-програми $\{I_1, I_2, I_3, \dots, I_n, I_1, I_2, I_3, \dots, I_m\}$ по суті є командами програми Q , у яких кожна команда вигляду $J(m, n, q)$ замінена командою $J(m, n, q + k)$.

МНР-програми P та Q **еквівалентні**, якщо при роботі над однаковими початковими конфігураціями вони або обидві зупиняються з однаковими фінальними конфігураціями, або обидві не зупиняються.

Приклад. МНР-програма для функції $f(x, y) = x + y$

Необхідні команди для обробки даної функції:

1. $J(1, 2, 5)$
2. $S(0)$
3. $S(2)$
4. $J(0, 0, 1)$

Продемонструємо роботу програми на конкретному прикладі $f(2, 3) = 2 + 3$

R_0	R_1	R_2	R_3
2	3	0	0
3	3	0	0
3	3	1	0
4	3	1	0
4	3	2	0
5	3	2	0
5	3	3	0

Результат в клітинці R_0 , що відповідає $2+3=5$

Демонстрація роботи МНР програми

Рисунок 4.3 – Сторінка МНР програми

Конфігурація, або повний стан МТ – це слово вигляду qx_1y , де $x, y \in \Sigma$. Неформально це означає, що на стрічці записане слово x_1y , тобто зліва і справа від x_1 можуть стояти тільки символи λ , МТ знаходиться в стані q , голівка чи *Машина Тюрінга* знаходиться в стані q_0 , де i -й та останній символи слова x відмінні від λ , назвемо початковою. Конфігурацію вигляду x_1y назвемо фінальною.

Приклад. МТ для обчислення функції $x - y$

```

q0l → q1λR
q1l → q1lR
q1# → q1#R
q1λ → q2λL
q2l → q3λL
q3l → q3lL
q3# → q3#L
q3λ → q0λR
q2# → q*|
q0# → q4λR
q4λ → q*λ

```

Демонстрація роботи МТ для функції $x - y$ при $x=2, y=1$

Введіть функцію: x-y
 Opened file 'x-y.mt'
 Встановлено обмеження в 10000000 кроків
 Введіть аргументи для функції через кому: 2,1
 =====
 [0]01110110[8]
 |1
 0:00 / 0:36 > q2 0 R

Рисунок 4.4 – Сторінка машини Тюрінга

Також для кожної сторінки реалізована спеціальна кнопка у вигляді стрілки вгору (розташована в правому нижньому куті), натискання на яку переміщує користувача на початок розділу.

4.2 Моделювання формальних моделей алгоритмів

Для моделювання роботи машини Тюрінга, було обрано мову програмування Python. На вхід програма приймає функцію, яка буде обчислюватись, і аргументи для функції. Функції, які доступні для моделювання: $x - y$, $x - 1$, $|x - y|$, $x * y$.

Після введення функції та аргументів для неї, програма виводить покрокове виконання заданої функції за допомогою машини Тюрінга (рисунок 4.5).

```

[0]01110110[8]
  |1
Code: q1 0 -> q2 0 R
  
```

Рисунок 4.5 – Частина виводу програми

В першому рядку виводиться пам'ять від 0 до 8 комірки. Символом «|» показана комірка пам'яті, де знаходиться каретка, число після цього символу – стан МТ. Наступна стрічка – код, який буде виконуватись.

Візуалізація роботи МНР програми було реалізована засобами HTML, CSS та JavaScript. Для початку потрібно задати необхідні інструкції та початкові стани регістрів (рисунок 4.6).



Рисунок 4.6 – Головна сторінка додатку

В результаті користувач отримує таблицю з результатом. Наведемо приклад для демонстрації роботи додатку, візьмемо функцію $x - y$, та аргументи 9 та 7 для неї (рисунок 4.7).

J	Z	T	S
1-J	1,2,6		
2-S	2		
3-S	3		
4-J	1,2,6		
5-J	1,1,2		
6-T	3,1		

Submit

9	7	0	0	0	0	0	0	0	0
R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
9	8	0	0	0	0	0	0	0	0
9	8	1	0	0	0	0	0	0	0
9	8	1	0	0	0	0	0	0	0
9	9	1	0	0	0	0	0	0	0
9	9	2	0	0	0	0	0	0	0
9	9	2	0	0	0	0	0	0	0
2	9	2	0	0	0	0	0	0	0

Рисунок 4.7 – Демонстрація роботи МНР програми

4.3 Вимоги до ПЗ

- для користування веб-додатком достатньо мати ОС з підтримкою веб-браузера;
- редактор коду або інтегроване середовище розробки (IDE) для розширення функціоналу або зміни інтерфейсу;
- процесор з достатньою потужністю для відкриття додатку у веб-браузері;
- доступ до мережі Інтернет.

ВИСНОВКИ

У ході виконання бакалаврської роботи було виконано наступні завдання:

- Проаналізовано та зроблено огляд технологій, які можуть бути використані для створення електронного посібника, встановлено їх переваги та недоліки.
- Зроблено аналіз та огляд дистанційного навчання, засобів та технологій, які вони використовують, а саме, інформаційні технології та мультимедійні засоби.
- Розроблена візуалізація формальних моделей алгоритмів (МНР програма та машина Тюрінга).
- Розроблено веб-додаток електронного посібника з курсу «Теорія алгоритмів».

На основі використання сучасних інструментів та технологій веб-розробки та встановлення їх переваг та недоліків було створено веб-проект, який включає в себе наступне:

1. Простота у використанні. Веб-проект розрахований на звичайного користувача, без необхідності поглиблених знань комп'ютерних технологій. Це дозволяє кожному користувачеві зрозуміти, як користуватися додатком.
2. Інтерфейс користувача. Розроблено зручний інтерфейс, в якому елементи на сторінці розміщені з врахуванням зручності та інтуїтивного розуміння для користувача. Додано навігаційне меню, яке розбиває посібник згідно до тем програми та дозволяє швидко та зручно переходити між ними.
3. Функціонал. Розроблений лише той функціонал, який потрібний для зручності користувача та відсутній зайвий функціонал, що може

завадити коректній роботі застосунку.

Таким чином, розроблений веб-проект електронного навчального посібника з курсу «Теорія алгоритмів» може бути використаний, як доповнення друкованого посібника. Використані технології – TypeScript та React, роблять додаток досить швидким та дають можливість доволі легко доповнити посібник новим матеріалом. Теми електронного посібника відповідають основним темам програмного курсу. Використання даного посібника дозволяє зменшити час на пошук інформації з навчальної дисципліни та покращує засвоєння матеріалу за рахунок використання мультимедійних технологій.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Поняття дистанційної освіти, особливості методів університетської освіти [Електронний ресурс] – Режим доступу до ресурсу: http://www.rusnauka.com/5_NITSB_2009/Pedagogica/41249.doc.htm
2. Рукавішнікова О. В., Теоретичні питання організації дистанційного навчання в освітній роботі [Електронний ресурс] – Режим доступу до ресурсу: <http://www.mcppv.ho.ua/docs/rukavishnikova01.htm>
3. Впровадження інформаційних технологій у навчальний процес [Електронний ресурс] – Режим доступу до ресурсу: <https://itcentres.lnu.edu.ua/e-learning/introduction-it-in-education/>
4. І.В. Ставицька, Інформаційно-комунікаційні технології в світі [Електронний ресурс] – Режим доступу до ресурсу: <http://confesp.fl.kpi.ua/ru/node/1103>
5. Волкова Т. Методичні засади створення і використання електронного підручника у професійній підготовці кваліфікованих робітників. Методичний посібник. – Київ: 2012. – 82 с.
6. Документація Visual Studio Code [Електронний ресурс] – Режим доступу до ресурсу: <https://code.visualstudio.com/docs>
7. Документація TypeScript [Електронний ресурс] – Режим доступу до ресурсу: <https://www.typescriptlang.org>
8. Документація React [Електронний ресурс] – Режим доступу до ресурсу: <https://reactjs.org>
9. М.С. Нікітченко, О.С. Шкільняк, С.С. Шкільняк. Теорія алгоритмів: навчальний посібник – Київ: Видавничо-поліграфічний центр «Київський університет», 2015. – 241с.

ДОДАТКИ

ДОДАТОК А.

Посилання на код веб-проєкту:

<https://github.com/illia1810/theory-of-algorithms-ts>