МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
 «ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «Системного аналізу та інформаційно-аналітичних технологій»

Оцінка

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

голова комісії

доц. каф.САІТ

\_\_\_\_\_\_\_\_\_\_\_\_ Марченко Н.А.

« » \_\_\_\_\_\_\_\_\_\_\_ 2017 р.

КУРСОВА РОБОТА

Дисципліна: «Чисельні методи»

Тема: « Численне розв’язування звичайних диференційних рівнянь»

Виконавець: ст. гр. КН-55г Брайко Ю.Г.

“\_\_ ”\_\_\_\_\_\_\_\_\_ 2017р.

Керівник роботи: доц., канд. техн. наук Г.Ю. Сидоренко

“ \_\_”\_\_\_\_\_\_\_\_\_ 2017 р.

Харків 2017

Зміст

[Вступ 3](#_Toc485686869)

[1. Постановка задачі 4](#_Toc485686870)

[2. Огляд існуючих методів вирішення задачі 5](#_Toc485686871)

[2.1 Метод Ньютона 5](#_Toc485686874)

[2.2 Метод простих ітерацій 5](#_Toc485686875)

[3. Чисельні методи вирішення 7](#_Toc485686876)

[4. Опис програми 9](#_Toc485686877)

[4.1 Керівництво користувача 9](#_Toc485686880)

[4.2 Опис розроблених об’єкті 10](#_Toc485686881)

[4.3 Порівняння результатів 11](#_Toc485686882)

[Висновок 15](#_Toc485686883)

[Список літератури 16](#_Toc485686884)

# Вступ

Потреба у розв’язку диференційних рівнянь часто зустрічаються, як пункти в алгоритмі вирішення, в багатьох задачах найрізноманітніших галузей, починаючи з фізики і закінчуючи прикладною соціологією. Оскільки в практичних задачах, зазвичай, складно занайти аналітичне рішення – числені методи набули широкого застосування. Вони дозволяють досягти універсальності рішень, з незначними жертвами точноісті результатів.

Окремо слід виділити клас карйових задач. Вони зустрічаються в різноманітних задачах, в котрих відомі початкові і кінцеві умови єксперементу і потрібно відновити закон, за яким власне проходив єксперимент. Отож не дивно що однією з перших галузей застосування методів подібного типу була балістика.

# Постановка задачі

Розробити програмне забезпечення, для вирішення наступної краєвої задачі:

*На проміжку [a,b], де a =0,b=1, d0=1.0, d1=4.0*

# Огляд існуючих методів вирішення задачі

## Метод Стрільби

Якщо звичайне диференціальне рівняння другого порядку – лінійне, то воно має вигляд:



при



Крайову задачу можна звести до задачі Коші введенням додаткової початкової умови, крім y(a)=A вводиться y΄(a)=1.

Знайшовши розв’язок y1(x), можна поставити іншу початкову умову y΄(a)=2 і отримати інший розв’язок y2 (x). Якщо а y2(b)=β2, причому β1 ≠β2 , то розв’язок:



буде задовольняти обидві початкові умови.

При розв’язуванні нелінійного звичайного диференціального рівняння методами “стрілянини” крайова задача зводиться до розв’язування декількох задач Коші, послідовно вводячи в початкові умови значення :

y(a)=A і y΄(a)=a

і намагаючись знайти розв’язок, який задовольняє умову y(b)=B.

При цьому алгоритм досягнення мети будується на основі одного з методів оптимізації. Однак цей шлях розв’язання задачі пов’язаний з великими обчислювальними труднощами, і тому у випадку нелінійних диференціальних рівнянь перевага надається різницевим методам.

## Метод прогонки

Перевага кінцево-різницевих методів в тому, що вони дозволяють звести розв’язок крайової задачі до розв’язку системи алгебраїчних рівнянь. При розв’язку двоточкової крайової задачі при і інтервал можна розділити на n рівних частин: де а В точках , що називаються вузлами, намагаються найти значення розв’язок . Знаючи координати вузлів користуючись кінцево-різницевими виразами для похідних





можна представити диференціальне рівняння у виді різницевого рівняння.

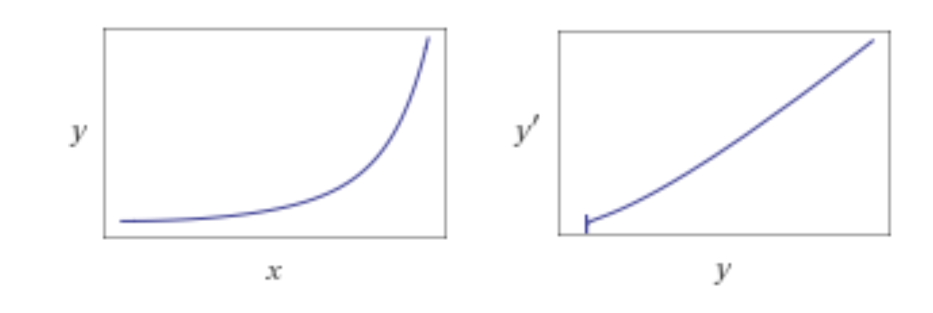
а) якщо початкове ЗДР лінійне, то задача буде складатися з розв’язання системи лінійних алгебраїчних рівнянь;

б) якщо початкове ЗДР нелінійне, то задача зводиться до розв’язування системи нелінійних алгебраїчних або трансцендентних рівнянь;

в) методи, алгоритми та програми розв’язування таких лінійних та нелінійних рівнянь відомі, але звести розв’язування крайової задачі методом кінцевих різниць до стандартної програми важко, оскільки формулювання кожної задачі залежить від вигляду рівняння, що розглядається.

# Чисельні методи вирішення

Для перевірки правильності використовувася сервіс wolframalpha з наступним запитом «y''+x\*cos(x)\*y'-y\*e^x=1+x;y(0)=1;y(1) = 4;». Результат відображений на рисунку 3.1

рис 3.1

# Опис програми

## Високорівнева архітектура

Програма представляє собою вебсервіс, котрий складається з серверної частини, написаної на Django(Python) та використовує в якості системи управління бази данних Postgres SQL. а в якості інструментів для рендру веб сторінки використовується Angular 4.0(Type script), HTML, CSS. Вибір наведеної структури обумовлений тим, що він дозволяє забезпечити роботу без прив’язки до конкретної платформи, адже єдиною умовою роботи данного рішення у кінцевого користувача є наявність веб браузера. Всі використанні інструменти доступні за лінцензіями, котрі не накладують обмеження на використання в начальних цілях.

## Інструкція користувавча

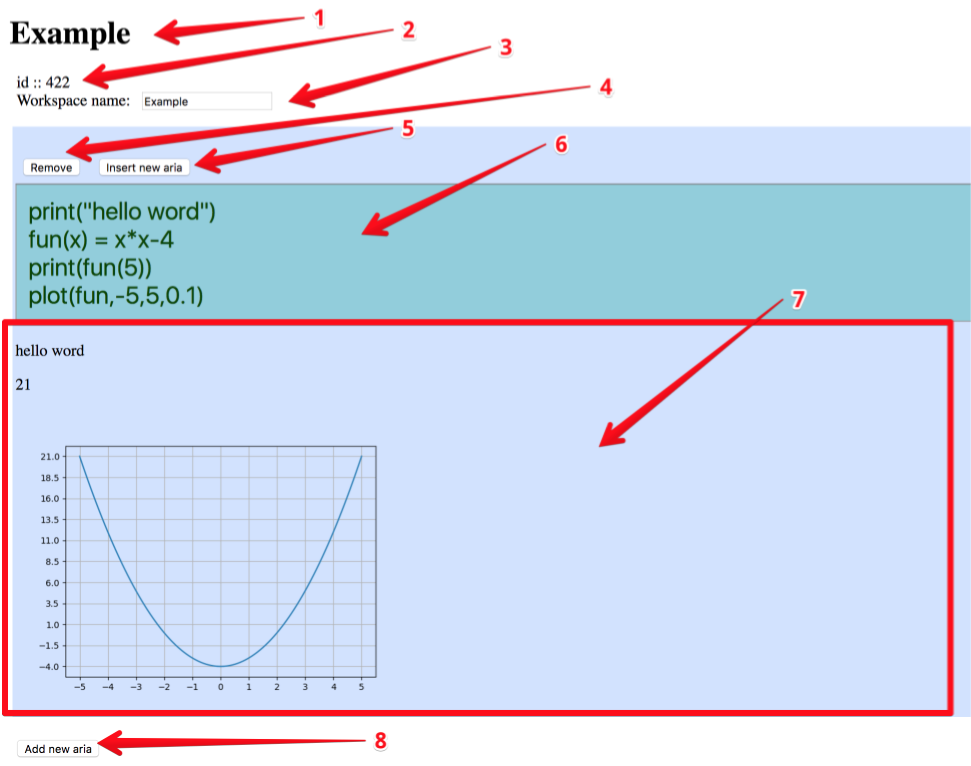


Рис 4.2.1

Для роботи потрібно відкрити адресу сервера, на якому розгорнутий веб додаток, після чого користувач побачить інтерфейс, зображений на рисунку 4.2.1. Елемент 1 являє собою назву робочого простору, його можно змінити за допомогою поля вводу 3. 2 елемент відображає индекс данного робочого простору в базі данних. Кнопка 4 видаляє данну робочу область а нопка 5 додає нову робочу облать попереду данної. Область 6 пердставляє собою вікно редактора в якому можна набирати математичні вирази, результат виконання яких буде відображенно в полі 6. Ото редактор дозволяє створювати власні змінні, функції, та вирішувати краєві задачі 2 порядку. Для виводу інформації використовуються функції print та plot. Перша приймає строку або ж число і друкує її в області виводу. Друга приймає посилання на функцію одного аргументу, 2 та 3 параметри представляють крайні значення графіку і остання – крок для отримання точок які будуть з’єднанні прямою для апроксимації істиної функції. Кнопка 8 дозволя додавати нове поле вводу після усіх існуючих полів.

Для розв’язку рівнянь слід використати наступну конструкцію - Eq.Diff.marginal(). Перший параметр приймає строку в якій записано диференційне рівняння вигляду “f`•`+a•f`+b•f+c=0”, де відповідно f – шукана функція та змінні a,b,c коефіцієнти або ж імена функцій від одного аргументу, котрі мають бути оголошені вище. Наступні 4 аргументи – це значення функції на краях. Отож дана функція повертає об’єкт, котрий репрезентує задане рівняння та містить інформацію про ім’я невідомох функції, вихідних код рівняння та обмеження. Для вирішення рівняння слід викликати у об’єкта рівння наступну функцію: calculate котра прийма ім’я метода для розвязку: «run» або «follow up». Метод calculate повертає функтор котрий містить іннформацію про рішення то може бути використаним, як функція на проміжку, котрий був визначений при заданні крайових умов.

Для кращої демонстрації введемо наступний текст скрипту в данну программу:

p(x) = x\*cos(x)

q(x) = -exp(x)

u(x) = 1+x

equation=Eq.Diff.marginal('y``+q(x)\*y+p(x)\*y`-u(x)',0,1,1,4)

print(equation.source)

print(equation.conditions)

functor = equation.calculate('follow up')

print('function name: '+functor.name)

print('approximation grid: ')

print( functor.\_grid)

plot(functor,0,1,0.1)

plot(equation.calculate('run'),0,1,0.1)

І отримає нустпуний текстовий результат:

y``+q(x)⋅y+p(x)⋅y`-u(x)

{'a': 0, 'fa': 1, 'b': 1, 'fb': 4}

function name: y

approximation grid:

{0: 1, 0.01: 1.0306020203689619, 0.02: 1.0462081031061317, 0.03: 1.0620203131134738, 0.04: 1.0780407395144755, 0.05: 1.0942714970199743, 0.06: …

Та графічний вивід зображений на рисунках 4.2.2 та 4.2.3, котрі представляють собою результат роботи для першого та другого способів відповідно.

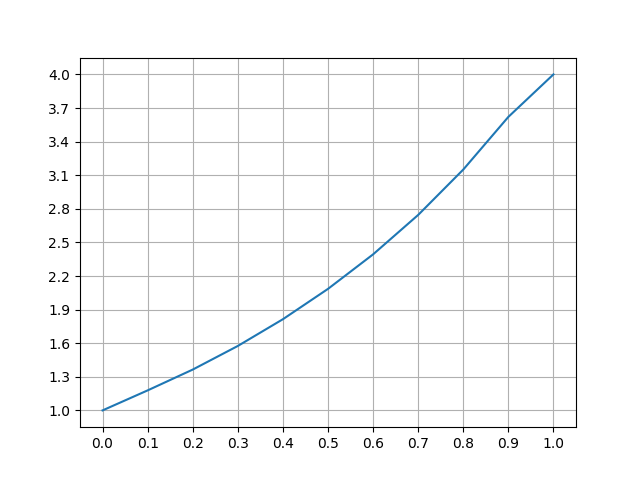


Рис 4.2.2

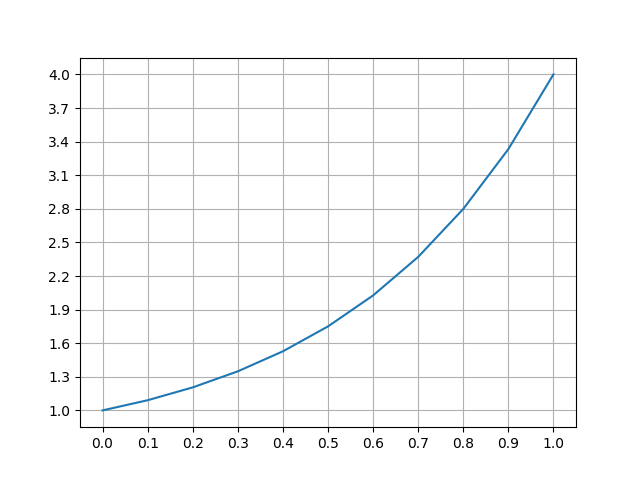


Рис 4.2.3

## Опис функціонування системи

Частина, котра відповідає за фронтенд фактично являє собою сутністність View та не містить суттєвої логіки. Основна її робота заключається в пересиланні користувацьких подій на сервер та оброблювати відповіді сервера, інтерпретуючи їх або ж системні події(додавання нових областей, зміна назви ітд) або ж відповідним чином редагуючи текст та зображення для відображення користувачу. Архітектура рішення продектована фреймворком Angular 4.

Серверна частина являє собою сервіс, котрий має певне API для взаємодії з стороніми сервісами. Відмова від використання вбудованих шаблонів для відображення інформації в користь REST архітектури була продиктована тенденції IT сфери до винесення візувального представлення в середовище з іншим стеком технологій та окремими командаим розробки(в конкретному випадку я займався обома частинами) Подібний підхід задовольняє вимогам «одна сутність – одна відповідальність», що дозволяє розробляти незалежний код, що як наслідок забезпечує більш гнучке рішення.

Інтерпретація скриптової мови відбувається шляхом трансляції її на мову пітон та виконання отриманого скрипту. Під час трансляції робляться в згенерований код додаются елементи для зберігання тексту та зображень(фактично передаються як текст, оскільки генеруються як svg).

Передбачаючи можливі шлях розширення інформація про робочий прострі зберігається в базі даних та репрезентуєтся в якості моделі. Подібним чином представлена і кожне, окреме поле вводу. Для забезпечення нормалізації бази данних поля вводу представлені окремою таблицею.

## Порівняння результатів

Результати роботи програми можна переглянути на рисунках 4.2.2 та 4.2.3 та порівнявши результати, отримані за допомогою wolframalpha. Оскільки в результаті було отримано схожі графіки то можна стверджувати що програма працює коректно.

# Висновок

Під час роботи над курсовим проектом я дослідив методи численного знаходження розв’язків диференційних рівнянь, а саме крайових задач. Для цього я розробив веб сервіс та користувацький односторінковий додаток та фласну скриптову мову що дозволить в інтерактивному режимі працювати з базовими математичними операціями та власне вирішувати вказані рівняння. Вихідний кодзнаходится за адресою <https://github.com/flying-pi/CourceWork3>.

Для перевірки коректності обчислень я порівнянв результати різних методів та користувалась сервісом wolframalpha.

# Список літератури

1. Каханер Д., Моулер К., Нэш С. Численные методы и программное обеспечение / Д. Каханер, К. Моулер, С. Неш. – М. : Мир, 2001.
2. Цегелик Г.Г. Чисельні методи / Г.Г. Цегелик. – Львів : ВЦ ЛНУ ім. І. Франка, 2004