

УДК 004.03

СТВОРЕННЯ ДОДАТКІВ В СЕРЕДОВИЩІ MAPLE

Бас Євгеній

Науковий керівник: канд. фіз.-мат. наук Гуртовий Ю.В.

*Центральноукраїнський державний педагогічний університет імені
Володимира Винниченка, м. Кропивницький, Україна*

Стаття присвячена загальним принципам створення графічних додатків в середовищі Maple. Наведено опис основних елементів таких додатків та створено маплет для знаходження кореня нелінійного рівняння методом поділу відрізка навпіл з наперед заданою точністю.

Ключові слова: Maple, маплет, чисельні методи.

Creating apps on Maple

Bas Yevhenii

Scientific supervisor: Yu.V. Hurtovyi, PhD

*Volodymyr Vynnychenko Central Ukrainian State Pedagogical University,
Kropyvnytskyi, Ukraine*

The article is devoted to the general principles of creating graphical applications in the Maple environment. Here is a description of the main elements of the application. A maplets was created to find the root of a nonlinear equation.

Keywords: Maple, maplet, numerical methods.

Постановка проблеми. Сучасне математичне моделювання – це потужна галузь науки, яка включає в себе необхідність використання різних умінь та навичок прикладного та теоретичного характеру. З розвитком потужностей комп'ютерної техніки вдається будувати та обробляти все складніші моделі навколишнього світу. Все це вимагає не лише володіння предметними областями, а і вміння активно та правильно використовувати сучасні досягнення в області математичних пакетів та програм. Сучасні CMS, такі як Maple, Mathematica, Maxima та інші, мають досить потужний набір інструментів для моделювання досить широкого кола явищ та процесів. Одним із основних напрямків в розвитку математичного моделювання є використання технологій візуального програмування та моделювання, які наявні в більшості сучасних CAS.

Аналіз досліджень і публікацій.

Сучасні прикладні дослідження, пов'язані з аналітичним та чисельним аналізом результатів потребують широкого застосування математичних пакетів різної спрямованості. Одним із найпопулярніших в застосування є пакет символічних обчислень та комп'ютерної графіки Maple.

В підручниках [1-3,8-12] наведено основні відомості про систему Maple, описано принципи роботи з цією системою, наведено приклади елементарного застосування в різних областях математики.

В монографіях [4,6,13] показано застосування системи Maple до розв'язання складних задач теоретичної фізики, зокрема механіки. Розглянуто приклади застосування цього пакету до розв'язування задач математичної фізики.

В [5] показано застосування CMS Maple до задач кінетичної хімії, та вказано на переваги застосування саме цього пакету в порівнянні з MathCad та Maxima. В [7] наведено приклад застосування системи maple до задач теорії графів з використанням маплетів.

Мета статті: показати можливості створення програмних додатків в середовищі Maple на прикладі створення маплету для реалізації методу дихотомії для відшукування коренів нелінійних рівнянь з наперед заданою точністю.

Основна частина.

Перед створенням будь-якого програмного додатку необхідно сформулювати технічне завдання, тобто вказати на можливості програми. Сформулюємо технічне завдання в наступному вигляді: створити програму в середовищі Maple для знаходження наближеного значення кореня нелінійного рівняння методом дихотомії з наперед заданою точністю. Програмний додаток повинен дозволяти:

- вводити рівняння, корінь якого необхідно знайти;
- будувати графік рівняння;
- вводити проміжок ізоляції кореня та точність;

- виводити наближене значення кореня для вказаної точності;
- мати зручний графічний інтерфейс.

Для створення програми використаємо можливості системи Maple для побудови графічних додатків або так званих маплетів. Розробка маплету складається з трьох основних етапів.

Етап перший – це розробка обчислювальної частини. Фактично на цьому етапі ми виконуємо чисельні розрахунки, будуємо шукані функції, створюємо графічні елементи тощо. Всі ці частини повинні бути виконані у вигляді процедур або функцій. Кожна процедура (функція) у Maple задається наступним чином:

*Im'я_процедури := **proc**(набір входних змінних)::вихідний тип даних
параметри процедури*

тіло (обчислювальна частина)

end proc:

Нами було створено дві процедури Grafik та Dihotomi. Перша процедура будує графік введеного рівняння, а друга призначена для знаходження наближеного значення кореня рівняння. Коди процедур представлені нижче.

Grafik := **proc** (Eqv)

local f;

f := unapply(lhs(Eqv)-rhs(Eqv), x);

plot(f(x), x, y)

end proc;

Dihotomi := **proc** (Eqv, Vidr, Toch)

local a, b, c, f, i, PKor;

f := unapply(lhs(Eqv)-rhs(Eqv), x); a := Vidr[1]; b := Vidr[2]; i := 0;

while Toch < evalf(b-a) **do**

c := (1/2)*a+(1/2)*b;

if f(c) = 0

then return evalf(c); **break**

else if evalf(f(a)*f(c)) < 0

```

    then b := c
    else a := c
end if
end if
end do;
return evalf(c)
end proc;

```

Другий етап – створення графічного інтерфейсу. На цьому етапі необхідно чітко визначити, які елементи будуть використані для побудови маплету, та в якому порядку вони будуть розташовані. Всі елементи для створення маплетів можна поділити на дві групи – елементи розмітки та функціональні елементи. перші слугують для створення графічної структури вікна маплету, другі відповідають за функціональну частину роботи маплету. Нижче наведено основні елементи розмітки (таблиця 1) та функціональні елементи(таблиця 2), які були використані нами для створення програми.

Таблиця 1.

Основні елементи розмітки

Назва елемента	Призначення
Window()	Генерація вікна маплету
BoxLayout()	Виділення однієї структурної робочої області
BoxColumn()	Виділення колонки в робочій області
BoxRow()	Виділення рядка в робочій області

Таблиця 2.

Основні функціональні елементи

Назва елемента	Призначення
Plotter()	Створення області для графічного об'єкту
TextField()	Створення текстового поля введення

Відмітимо, що на другому етапі важливим є правильне розташування всіх елементів у тому порядку, який передбачений розробником. В нашому випадку графічний інтерфейс програми представлено на мал. 1.

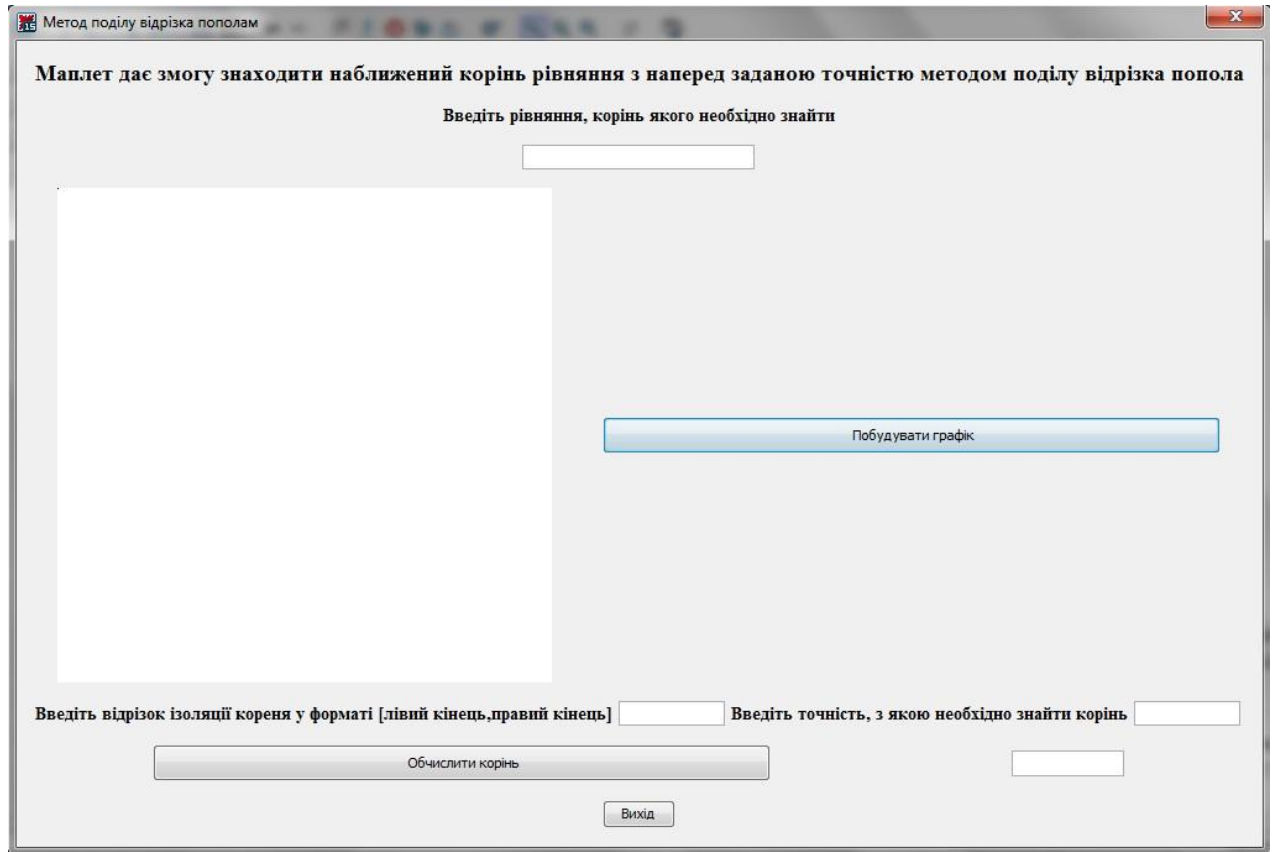


Рис. 1. Зовнішній вигляд створеного маплету

Третій етап – поєднання функціональних елементів та обчислювальної частини. На цьому етапі ми програмуємо всі створені функціональні елементи, спів вставляючи їм раніше створені обчислювальні процедури тощо. Покажемо цей процес на прикладі програмування кнопки «Обчислити корінь»:

`Button(" Обчислити корінь", Evaluate('TF4' = 'Dihotomi(TF1, TF2, TF3)').`

Виконання цього коду означає: створити кнопку з назвою «Обчислити корінь» та, при натисканні на неї виконати процедуру `Dihotomi`, якій в якості параметрів передати введені користувачем рівняння, відрізок ізоляції кореня та точність.

Висновки та перспективи подальших пошуків у напрямі дослідження

Використання візуального програмування в середовищі Maple для математичного моделювання реальних явищ та процесів має ряд переваг, порівняно з використанням інших мов програмування, як то C, Python та ін.

Більшість задач, пов'язаних з математичним моделюванням призводить до дослідження диференціальних рівнянь та систем таких рівнянь, в тому числі і в частинних похідних. Maple має вже готові вбудовані функції для роботи з такими математичними об'єктами, і, що важливо, в символному вигляді. Це значно полегшує роботу, особливо у випадку наявності точних аналітичних або наближених аналітичних розв'язків. При аналізі математичних моделей дуже часто доводиться використовувати якісні методи дослідження, пов'язані з візуалізацією отриманих числових або аналітичних даних. Система Maple має потужний інструментарій для створення та обробки візуальної інформації (побудова двомірної та тривимірної графіки тощо). Створення візуальних середовищ для дослідження математичних моделей дає можливість вивчати поведінку досліджуваного явища чи процесу для багатьох наборів вхідних параметрів, що вирішується за допомогою створення мап летів, які по суті є користувацьким інтерфейсом при роботі з конкретними моделями.

Список використаної літератури

1. Савотченко С.Е. «Методы решения математических задач в Maple: Учебное пособие» / Савотченко С.Е., Кузьмичева Т.Г. – Белгород: Изд. Беллаудит, 2001. – 116 с.
2. В.З. Аладьев, В.К. Бойко, Е.А. Ровба "Программирование в пакетах Maple и Mathematica: Сравнительный аспект" / Монография / Гродно: Гродненский Госуниверситет, 2011, 517 с.
3. М. Н. Кирсанов. "Практика программирования в системе Maple" М.: Издательский дом МЭИ, 2011, 208с.
4. М. Н. Кирсанов. Задачи по теоретической механике с решениями в Maple 11. М.: Физматлит, 2010, 264с.
5. В.И. Коробов, В.Ф. Очков. Химическая кинетика: введение с Mathcad/Maple/MCS. М.: Горячая линия-Телеком, 2009.
6. Чарльз Генри Эдвардс , Дэвид Э. Пенни. Дифференциальные уравнения и краевые задачи: моделирование и вычисление с помощью Mathematica, Maple и MATLAB. 3-е издание. Киев.: Диалектика-Вильямс, 2007. ISBN 978-5-8459-1166-7.

7. М. Н. Кирсанов. Графы в Maple. М.: Физматлит, 2007, 168с.
8. В.З. Аладьев, В.К. Бойко, Е.А. Ровба. Программирование и разработка приложений в Maple. Гродно, Таллин, 2007.
9. В.З. Аладьев. Основы программирования в Maple. Таллин, 2006.
10. Е. Р. Алексеев, О. В. Чеснокова. Решение задач вычислительной математики в пакетах Mathcad 12, MATLAB 7, Maple 9. М: НТ Пресс, 2006, 496с. ISBN: 5-477-00208-5.
11. В.З. Аладьев. Системы компьютерной алгебры. MAPLE: Искусство программирования. Бинوم.Лаборатория знаний, 2006.
12. В.П. Дьяконов. Maple 9 в математике, физике и образовании. М.: СОЛОН-Пресс, 2004.
13. Д.П. Голоскоков Уравнения математической физики. Решение задач в системе Maple. С-Пб: Питер, 2004.