Лабораторна робота № 4

Тема роботи: . Програмування у середовищі MatLAB

**Тема роботи:** Використання методу функції функцій та при підпрограм – функцій при програмування у середовищі MatLAB.

**Мета роботи**: Вивчення методів функції функцій та при підпрограм – функцій при програмування у середовищі MatLAB.

Постановка задачі: Створити програму обчислень з використання процедур – функцій у середовищі MatLAB.

Вступ

Робота у режимі калькулятора в середовищі MatLAB, незважаючи на досить великі можливості, має суттєві незручності:

– неможливо повторити попередні обчислення i дії для нових значень вхідних параметрів без повторного набирання попередніх операторів;

– неможливо повернутися назад i повторити деякі дії, або за деякою умовою перейти до виконання іншої послідовності операторів.

Тому складні, з перериваннями, складними переходами за деяких умов, з багаторазово повторюваними однотипними діями обчислення, які до того ж потрібно проводити неодноразово за інших вхідних даних, потребують їх спеціального оформлення у вигляді записаних на диску файлів, тобто у вигляді програм. Перевага програм у тому, що, завдяки їх зберіганню у вигляді записаних файлів, стає можливим неодноразове звернення до тих самих операторів i до програми.

Створення програм дозволяє значно скоротити процес підготовки повторюваних обчислень, зробити процес обчислень більш наочним i прозорим, а завдяки цьому різко зменшити ймовірність появи принципових похибок .при побудові процесу обчислень. Kpiм того, у програмах з'являється мoжливicть автоматизувати також i процес змінювання значень вхідних параметрів у діалоговому режимі.

1.1. Функції функцій

Деякі важливі процедури у MatLAB потребують при зверненні вказання iмені М-файлу, в якому міститься програма обчислення деякої iншoї процедури (функції). Такі процедури називають функціями функцій. Щоб скористатися такою функцією від функції для деякої необхідної користувачеві функції, треба, щоб попередньо користувач утворив М-файл, в якому би обчислювалося значення потрібної функції за заданим значенням її аргументу.

Наведемо приклад деякої функції від функції.

Побудова графиків функції oднієї змінної може бути здійснена за допомогою процедури fplot. Відмінність цієї процедури від plot у тому, що для побудови графіка функції нема потреби здійснювати попереднє обчислення значень функції. Звернення до неї має вигляд:

fplot ('<ім'я функції>, [<інтервал>], n),

де <інтервал> - це вектор з двох чисел, що вказують, відповідно, нижню i верхню мeжi змінювання аргументу; <ім'я функції > - ім'я М-файлу, де міститься процедура обчислення значення бажаної функції за заданим значенням її аргументу; n - бажана кількість частин розбиття вказаного інтервалу. Якщо останню величину при зверненні не вказано, за замовчуванням інтервал розбивається на 25 частин.

Якщо звернутися до цієї процедури таким чином:

[х, Y] = fplot ('<ім'я функції>, [<інтервал>], n);,

то графік вказаної у М-файлі функції не відображується на екрані (у графічному вікнi). Замість цього обчислюються вектор "х" аргументів та вектор (або матриця) Y відповідних значень вказаної функції. Щоб при зверненні останнього виду побудувати графік, необхідно виконати це у подальшому за допомогою процедури plot(x, Y).

і.2. Утворення М-файлів у середовищі MatLAB

1.2.1. Особливості утворення М-файлів у MatLAB

Утворення програми у середовищі MatLAB, через те, що власного "вбудованого" текстового редактора система MatLAB не має, здійснюється

за допомогою стороннього текстового редактора, який автоматично викликається, якщо його попередньо встановлено за допомогою команди "Editor Preference" з розділу "Options" головного меню командного вікна MatLAB. Наприклад, це може бути редактор Notepade середовища Windows. Вікно цього попередньо встановленого редактора з'являеться на екрані, якщо перед цим використано команди "M-file" з підщрозділу "New"

або обрано назву одного з існуючих М-файлів у підрозділі "Open M-file" розділу "File" меню командного вікна. В першому випадку вікно текстового редактора є порожнім, у другому - в ньому міститься текст викликаного файлу. В обох випадках вікно текстового редактора готове для введення нового тексту чи коригування існуючого.

Програми у мові MatLAB мають два piзнoвиди - так званий Script-файл та файл-функція. Обидва різновиди повинні мати розширення .m при запису тексту файлу на диск, тобто не можуть бути poзpiзнені за виглядом iмені.

За допомогою Script-файлів оформлюються головні програми, які керують з початку до кінця opгaнiзaцiєю усього обчислювального процесу,

та окремі частини головних програм (вони можуть бути записані у вигляді

окремих Script-файл. Як файл-функції оформлюються окремі процедури i функції (тобто такі частини програми, які poзpaxoвaнi на неодноразове використання Script-файлами або іншими процедурами при змінених значеннях вхідних пapaмeтpiв i не можуть виконуватися без попереднього встановлення значень деяких змінних, які називають вхідними).

Головною зовнішньою відмінністю цих двох видів файлів є те, що файли-функції мають перший рядок вигляду

function <ПКВ>=<ім'я процедури>(<ПВВ>)

де позначено ПКВ - Перелік Кінцевих Величин, ПВВ - Перелік Вхідних Величин. Script-файли такого рядка не мають.

Принципова ж відмінність полягає у зовсім іншому сприйманні системою iмен змінних у цих двох видах файлів. У файлах-функціях yci імена змінних всередині файлу, а також iмeнa змінних, вказані у заголовку (ПКВ та ПВВ), сприймаються як локальні, тобто yci значення цих змінних після завершення роботи процедури зникають i область оперативної пам'яті ПК, яку було відведено під запис значень цих змінних, звільняється для запису у ній значень інших змінних.

У Script-файлах yci використані змінні утворюють так званий "робочий npocтip (глобальний)". Значення i зміст їx зберігається при переході від виконання одного Script-файлу до виконання іншого, який викликано з першого, як i при поверненні до першого, тобто робочий пpoстip є єдиним для ycix Script-файлів. Саме завдяки цьому будь який довгий Script-файл можна розбити на декілька фрагментів, оформити кожен з них у вигляді окремих Script-файлів, а у головному Script-файлі замість відповідного фрагмента записати оператор виклику Script-файлу, що презентує цей фрагмент. Цим забезпечується компактне i наочне подання навіть досить складної програми.

За виключенням вказаних відмінностей, файли-функциї i Script-файли оформлюються однаково.

і.2.2. Головні особливості оформления М-файлів

У подальшому під М-файлом розумітимемо будь-який файл (файл-

функщю чи Script-файл), який записано мовою системи MatLAB.

Розглянемо головні особливості запису тексту програми (М-файлу)

мовою MatLAB.

і. Звичайно кожний оператор записується на окремому рядку тексту програми. Ознакою кінця оператора є символ повернення каретки (він не друкується) i переходу на наступний рядок, який вводиться у програму при натисканні клавіші <Enter>, тобто при переході при запису тексту програми на наступний рядок.

2. Можна розміщувати - кілька операторів у одному рядку. Тоді попередній оператор цього рядка повинен закінчуватися символом ";" або"," .

3. Можна довгий оператор записувати у кілька рядків. При цьому попередній рядок оператора має закінчуватися трьома крапками ("...").

4. Якщо черговий оператор не закінчується символом ";", результат його дiї при виконанні програми буде виведений у командне вікно. Щоб запобігти виведенню на екран результату дії чергового оператора програми, запис цього оператора у тексті програми повинен закінчуватися символом ";".

5. Рядок програми, що починається з символу "%", не виконуєгься. Цей рядок сприймається системою MatLAB як коментар. Таким чином, для введення коментаря у будь яке місце тексту програми, досить почати відповідний рядок з символу "%".

6. Рядки коментаря, які передують першому оператору програми, що виконуєгься (тобто не є коментарем), сприймаються системою MatLAB як

опис програми. Саме ці рядки виводяться у командне вікно, якщо у останньому набрано команду help <м'я файлу>.

8. У програмах мовою MatLAB немає символу кінця тексту програми.

9. У мовi MatLAB змінні не описуються i не оголошуються. Будь-яке нове ім'я, що з'являється у тексті програми сприймається системою MatLAB як ім'я матриці з комплексними елементами. Розмір цiєї матриці встановлюється попереднім введенням значень її елементів або визначається діями по встановленню значень її елементів, які описані у попередньому операторі або процедурі. Ця особливість робить мову MatLAB вельми привабливою i простою. У мовi MatLAB неможливе використання вхідної матриці або змінної, в якій попередньо не введені або обчислені значення її елементі (а тому - i визначені розміри цієї матриці). Інакше при виконанні програми MatLAB з'явиться повідомлення про похибку - "Змінна не визначена".

і0. Імена змінних можуть містити букви латинської абетки або цифри i повинні починатися з букви. Загальна кількість символів в імені може бути

до 19. В іменах змінних можуть використовуватися як великі, так i малі букви. Особливістю мови MatLAB є те, що великі i малі букви в іменах розрізнюються. Наприклад, символи «а» та «А» можуть використовуватися в одній nporpaмi для позначення різних величин.

і,3. Створення найпроспших файлів-функцій (процедур)

1.3.1. Загальні вимоги до побудови

Як зазначалося раніше, файл-функція (процедура) повинен починатися з рядка

function [<ПКВ>] = <ім'я процедури>(<ПВВ>).

Якщо перелік кінцевих (вихідних) величин (ПКВ) містить лише один об'єкт (у загальному випадку матрицю), то файл-функція репрезентує звичайну функцію (одної чи кількох змінних). Фактично навіть у цьому випадку файл-функція є процедурою у звичайному розумінні інших мов програмування. Перший рядок у цьому випадку має вигляд:

function <ім'я змінної> = <ім'я процедури>(<ПВВ>).

Якщо ж у результаті виконання файл-функції має бути визначені (обчислені) кілька об'єктів (типу матриць), цей файл-функція репрезентує вже більш складний об'єкт, який у програмуванні звично називається або процедурою (у мові Паскаль), або підпрограмою. Загальний вигляд першого рядка при цьому матиме вигляд:

function [yl, у2,..., yN] = <ім'я процедури>(<ПВВ>),

тобто перелік кінццевих величин yl, у2, ..., yN повинен бути поданий як вектор з елементами yl, у2,..., yN (кожен з яких може бути матрицею).

У найпростішому випадку функції однієї змінної перший рядок набуває вигляду:

function у =func(x),

ду func - iм'я функци (М-файлу).

Як приклад розглянемо складання М-файлу для функції

у=fі(х)= а\* sin(x)\* .

Для цього слід активізувати розділ "File" головного меню командного вікна MatLAB, обрати у підменю, що з'явиться на екрані, розділ "New", а потім - команду "M-file". На екрані з'явиться вікно текстового редактора. В ньому слід набрати наступний текст:

function у = Fl(x,а)

% Процедура, що обчислює значення функції

% у = а\*sin(x)\*х^(2/3)

% Звернення у = Fl(x,а).

у = а\*sin(x) . \*х^(2/3);

Після цього необхідно зберегти цей текст під iм'ям «Fl.m». Необхідний М-файл утворено. Відтепер можна користуватися цією функцією при розрахунках.

Так, якщо ввести команду

» у = Fl(pi/4, і),

то одержимо результат

у= 0.60193.

Слід зауважити, що аналогічно можна одержати одразу вектор ycix значень означеної функції за різних значень аргументу, якщо останні зібрати у деякий вектор. Так, якщо сформувати вектор

» zet=[ і; pi/4 ; 3.і4];

i звернутися до цієї ж процедури

» my = Fl(zet,l),

то одержимо:

my =

0.84147

0.60193

0.0034151

Примітки.

1. Можливість використання сформованої процедури як для окремих чисел, так i для векторів i матриць зумовлено застосуванням у запису відповідного М-файлу замість звичайних знаків арифметичних дій їх аналогів з попередньою крапкою.

2. Щоб уникнути виведення на екран небажаних проміжних результатів, необхідно у тексті процедури yci обчислювальні оператори завершувати символом ";".

3. Як показують наведені приклади, імена змінних, які вказуються у заголовку файл-функцій можуть бути якими завгодно, збігатися чи ні з іменами, що використовуються при зверненні до цієї файл-функції, тобто носять формальний характер. Важливіше за все лише те, щоб структура звернення повністю відповідала структурі заголовка у запису тексту М-файлу i щоб зміннi у цьому зверненні були того ж типу i розміру, що й у заголовку М-файлу.

Щоб одержати інформацію про створену процедуру, достатньо набрати у командному вікні команду:

» help fl,

i у командному вікні з'явиться:

Процедура, що обчислює значення функції

у = а\*sin(x)\*х^(2/3)

Звернення у = Fl(x,а).

**Завдання**

Завдання 1.1. Створити процедуру функцію обчислення суми нескінченного ряду з заданою точністю ε. Визначити кількість членів, необхідних для досягнення заданої точності ε. Створити програму, що за допомогою створеної підпрограми. виводіть до командного вікна таблицю результатів обчислень для різних значень параметру ε=10-2 ;10-3; 10-4 ;10-5 . В.аріанти вибрати з таблиці:

| №  вар. | Загальний член ряду | Х |
| --- | --- | --- |
| і |  | 0,12 |
| 2 |  | 0,811 |
| 3 |  | 0,54 |
| 4 |  | 0,21 |
| 5 |  | 0,27 |
| 6 |  | 0,85 |
| 7 |  | 0,29 |
| 8 |  | 0,72 |
| 9 |  | 0,19 |
| і0 |  | 0,64 |
| 11 |  | 0,56 |
| 12 |  | 0,63 |
| 13 |  | 0,4 |
| 14 |  | 1,2 |
| і5 |  | 0,8 |

**Запитання**

1. Задля чого створюються програми у середовищіц MatLAB?

2. Чим відрізняються файл-функції від Script файлів? Яка сфера

застосування кожного з цих видів файлів?

3. Що таке функції функцій?

4. Як створити М-файл процедури або функції?

5. Які головні правила написання текстів М-файлів у мовi MatLAB?

Література

1. Ю.Ф. Лазарев. Початки програмування у середовищі *MatLab* /Навчальний посібник. –К.: Корнійчук, 1999. – 160 с.