



ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
УПРАВЛЕНИЕ ДИСТАНЦИОННОГО ОБУЧЕНИЯ И ПОВЫШЕНИЯ
КВАЛИФИКАЦИИ

Кафедра «Программное обеспечение вычислительной
техники и автоматизированных систем»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

**"Инкрементная разработка простых графических
приложений средствами MATLAB.
Двумерный случай"**

по дисциплине

"Методы и системы компьютерной математики"

Автор

Коледов Леонид Викторович, к. ф.-м.н., доцент

Ростов-на-Дону, 2013



Оглавление

ЛАБОРАТОРНАЯ РАБОТА 1.....3

[Краткая теория](#)



ЛАБОРАТОРНАЯ РАБОТА 1

Тема: Инкрементальная разработка простого графического приложения средствами MATLAB.

Цель работы: знакомство с графическими функциями среды, приемами отладки, тестирования и документирования.

Задание 1. Ознакомьтесь с текстом `fplot1.m` прототипа программы, которую надлежит разработать. Для исследования различных возможностей используйте отладочные скрипты `fplots.m` и `fplots1.m`. В отчет включите 3-4 сохраненных графика с пояснениями, указывающими код их получения.

Задание 2. Напишите и отладьте программу-функцию `fplot01.m` функциональность которой ограничивается следующим.

`FPLOT01(FNAME,LIMS)` чертит график функции, специфицированной текстовой переменной `FNAME` в пределах, специфицированных по оси `x`-ов посредством параметра `LIMS = [XMIN XMAX]`.

Например, `FPLOT('sin',[0 10])` это график функции "sin" на отрезке `[0,10]`.

`FPLOT01(FNAME,LIMS,N)` чертит функцию на сетке минимальной длины `N`.

По умолчанию `N = 25`.

Проведите исследование на скрипте, подобном вышеприведенным. В отчет включите 3-4 сохраненных графика с пояснениями, указывающими код их получения.

Задание 3. Напишите и отладьте программу-функцию `fplot02.m` функциональность которой ограничивается следующим.

`FPLOT02(FNAME,LIMS)` чертит функцию специфицированную текстовой переменной `FNAME` в пределах, специфицированных по оси `x`-ов посредством параметра `LIMS = [XMIN XMAX]`.

`FPLOT02(FNAME,LIMS,N)` чертит функцию на сетке минимальной длины `N`.

По умолчанию `N = 25`.

`FPLOT02(FNAME,LIMS,N,ANGLE)` чертит график на сетке минимальной длины `N` и более часто, если отдельные сегменты графика разделены углом более `ANGLE` (в градусах). По умолча-



Методы и системы компьютерной математики

нию $\text{ANGLE} = 10$ градусов. `FPLOTT02(FNAME,LIMS,N,ANGLE,SUBDIV)` чертит график на сетке минимальной длины N , с дугами не более ANGLE градусов, но проводя не более, чем SUBDIV итераций, чтобы заполнить быстро меняющиеся сегменты.

По умолчанию $\text{SUBDIV} = 2$

Проведите исследование на скрипте, подобном вышеприведенным. В отчет включите 3-4 сохраненных графика с пояснениями, указывающими код их получения.



Функция fplot1.m

```
function [x0,y0] = fplot1(fname,lims,npts,angl,subdiv)
```

```
%FPLOTT Plot function.
```

```
% FPLOTT(FNAME,LIMS) plots the function specified by the text variable
```

```
% FNAME between the x-axis limits specified by LIMS = [XMIN XMAX].
```

```
% For example, FPLOTT('sin',[0 10]) is a plot of the sine function.
```

```
% FPLOTT(FNAME,LIMS,N) plots the function with a minimum of N samples.
```

```
% The default value for N is 25.
```

```
% FPLOTT(FNAME,LIMS,N,ANGLE) plots the function with a minimum of N
```

```
% samples and samples more if contiguous segments are at an angle
```

```
% larger than ANGLE (in degrees). The default for ANGLE is 10 degrees.
```

```
% FPLOTT(FNAME,LIMS,N,ANGLE,SUBDIV) plots the function with a minimum
```

```
% of N samples, no more than ANGLE degrees bends in the curve, but no
```

```
% more than SUBDIV iterations to fill in the rapidly changing portions.
```

```
% The default for SUBDIV is 20.
```

```
% [X,Y] = fplot(FNAME,LIMS,...) returns the abscissae and ordinates for
```

```
% FNAME in the column vectors X and Y. Plot X versus Y to see the picture.
```

```
% FPLOTT(FNAME,LIMS) чертит функцию специфицированную текстовой  
переменной
```

```
% FNAME в пределах, специфицированных по оси x-ов посредством парамет-  
ра
```

```
% LIMS = [XMIN XMAX].
```

```
% Например, FPLOTT('sin',[0 10]) это график функции "sin".
```

```
% FPLOTT(FNAME,LIMS,N) чертит функцию на сетке минимальной длины N.
```

```
% По умолчанию N = 25.
```

```
% FPLOTT(FNAME,LIMS,N,ANGLE) чертит график на сетке минимальной длины N
```

```
% и более часто, если отдельные сегменты графика разделены углом более
```

```
% ANGLE (в градусах). По умолчанию ANGLE = 10 градусов.
```

```
% FPLOTT(FNAME,LIMS,N,ANGLE,SUBDIV) чертит график на сетке минимальной
```

```
% длины N, с дугами не более ANGLE градусов, но проводя не более, чем
```

```
% SUBDIV итераций, чтобы заполнить быстро меняющиеся сегменты.
```

```
% По умолчанию SUBDIV 20.
```

```
% [X,Y] = fplot(FNAME,LIMS,...) returns the abscissae and ordinates for
```

```
% FNAME in the column vectors X and Y. Plot X versus Y to see the picture.
```

```
% L. Shure, 12-22-88
```

```
% (c) Copyright 1988, by The MathWorks, Inc.
```

```

if nargin < 2 % Если менее 2-х вх аргументов
    error('The X-axis limits must be specified for FPLOТ.');
```

% error('Пределы по оси X-ов должны быть указаны для FPLOТ.');

```

elseif nargin < 5 % Если менее 5-ти
    subdiv = 20; % Установить число итераций 20
    if nargin < 4 % Если менее 4-х вх аргументов
        angl = 10; % Установить угол 10 град
    end
    if nargin < 3 % Если менее 3-х вх аргументов
        npts = 25; % Установить число промежуточных точек
    end
end
end
iter = 0;
angl = angl*pi/180;
angl = abs(angl);
ang = angl + 1;
xmin = min(lims);
xmax = max(lims);
x = xmin + (0:npts-1)*(xmax-xmin)/(npts-1); % Массив значений аргумента
y = feval(fname,x); % Соответствующий массив значений функции
yscal = max(abs(y)); % Размер массива значений функции
y = y/yscal; % Развернем значения y-ов в [-1,1]

% see if need to fill in values for 'smoothness'
% Проверим, не требуется ли добавить значения для 'гладкости'
while any(ang > angl) & (iter <= subdiv) % Пока
    yi = diff(y); % Массив приращений y-ов
    xi = diff(x); % Массив приращений x-ов
    nx = length(xi) - 1; % Число отрезков для графика - 1
    radi = sqrt(xi.^2+yi.^2); % Массив длин этих отрезков
    num = xi(1:nx).*xi(2:nx+1) + yi(1:nx).*yi(2:nx+1); % Числители
    % т.е. скалярные произведения векторных приращений на соседних
    % интервалах
    den = radi(1:nx).*radi(2:nx+1); % Знаменатели
    % т.е. произведения модулей этих приращений
    ang = abs(acos(num./den)); % Массив поворотов этих приращений
    iter = iter + 1;
    ii = find(ang > angl); % need to interpolate more
    % Где найден ли поворот более 10 гр
    if isempty(ii)
        break;
    end
end

```



Методы и системы компьютерной математики

```

% Если найден, то добавить точки в середины интервалов
% для промежуточных значений, когда требуется
else
    xt = (x(ii) + x(ii+1))/2;
    % Массив промежуточных значений аргумента
    yt = feval(fname,xt)/yscal;
    % Соотв массив промежуточных значений функции, нормированный в
    % [-1,1]
    [x, ind] = sort([x;xt]);
% Вставим полученные промежуточные значения xt
% в массив ранее полученных аргументов
    y = [y;yt];
    y(:) = y(ind);
    % Соответственно добавим значения функции
end
end
y = y*yscal;
if iter >= subdiv
    disp('WARNING: Maximum number of iterations has been reached.');
```

if nargin == 0

```

    disp('      Plot may be inaccurate.');
```

end

end

if nargin == 0

```

    plot(x,y)
```

else

```

    x0 = x;
    y0 = y;
```

end

—

Функция g.m

```

function out=g(x)
% Обращение y=g(x) приводит к матрице размера size(x),
% компоненты которой sin(1/x)
out=x.*sin(1./x);
end
```



Функция dg.m

```
function out=dg(x)
% Обращение y=dg(x) приводит к матрице размера size(x),
% компоненты которой значения производной от функции x*sin(1/x)
out=sin(1./x)-cos(1./x)./x;
end
```

Скрипт fplots.m

```
clear all
x1=0.1:0.01:0.3;
x2=0.01:0.001:0.03;
x3=0.001:0.0001:0.003;

figure(1),fplot1('g',[1,3]),grid
figure(2),fplot1('g',[0.01,3]),grid
figure(3),fplot1('dg',[0.01,0.03]),grid
figure(4),fplot('dg',[0.001 0.003]),grid
pause

figure(5),
subplot(3,1,1),plot(x1,g(x1)),grid
subplot(3,1,2),plot(x2,g(x2)),grid
subplot(3,1,3),plot(x3,g(x3)),grid

figure(6)
subplot(3,1,1),fplot1('g',[0.1 0.3]),grid
subplot(3,1,2),fplot1('g',[0.01 0.03])
subplot(3,1,3),fplot1('g',[0.001 0.003]),grid
pause

figure(7)
subplot(3,1,1),fplot1('dg',[0.1 0.3]),grid
subplot(3,1,2),fplot1('dg',[0.01 0.03]),grid
subplot(3,1,3),fplot1('dg',[0.001 0.003]),grid
pause

figure(8)
```




```
subplot(3,1,1),fplot1('g',[0.1 0.3],100,1000),grid  
subplot(3,1,2),fplot1('g',[0.01 0.03],100,1000)  
subplot(3,1,3),fplot1('g',[0.001 0.003],100,1000),grid
```

```
*****
```

Скрипт **fplots1.m**

```
*****
```

```
fplot1('g',[0.0001 0.0003],100,1000),grid  
figure(2),fplot1('g',[0.0001 0.0003],100,1000),grid  
figure(3),fplot1('g',[0.0001 0.0003],1000,1000),grid  
figure(4),fplot1('g',[0.0001 0.0003],10000,1000),grid  
figure(5),fplot1('g',[0.0001 0.0003],100000,1000),grid  
figure(6),fplot1('g',[0.0001 0.0003],1000000,1000),grid  
figure(7),fplot1('g',[0.0001 0.000103],1000000,1000),grid
```