

作业 #1

课程: XX 课程

姓名: 学生名字 专业: 专业名称 学号: 123002584

日期: 2024 年 8 月 22 日

问题 1. 这里是一个问题.

解: 这里是问题的解答.

问题 2. 这里是一个证明问题.

证明: 这里是问题的证明.

□

这里定义了一个新的盒子环境 `hframe`, 它是没有编号的. 使用者可以写问题, 或者答案, 或者其他任何内容.

1. 第一项
2. 第二项
3. 第三项

定理环境

定义 1. 这是一个定义.

命题 1. 这是一个命题.

引理 1 (Lemma). 这是一个引理.

定理 1 (Theorem). 这是一个定理.

证明: 这是证明环境.

□

推论 1. 这是一个推论.

命题 2 (Proposition). 这是一个命题.

注 1. 这是一个 *remark*.

例 1. 这是一个例子.

定理 A.1. 这是一个自定义的定理.

表格

这是一个表格示例, 如表 1

表 1 表格名字

A	$N = 3$	$N = 5$	$N = 7$	$N = 9$	$N = 11$	$N = 13$
B	1.5789	1.3478	1.0645	0.8780	0.7222	0.5942
C	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
D	7.2632	14.3913	21.0323	27.3171	30.9630	34.0870

图片

这是一个插图示例, 如图 2.

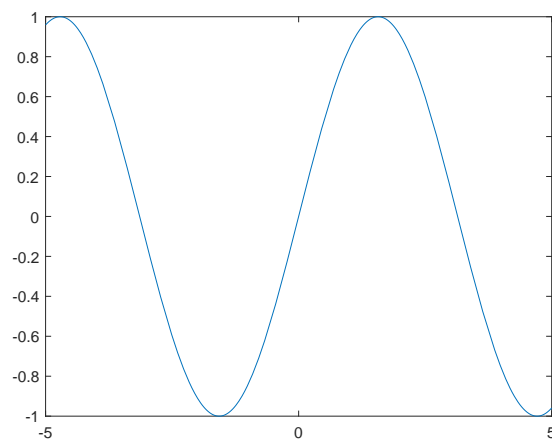


图 1 函数 $y = \sin(x)$ 的图像

这是一个两个图片并排放置的示例, 如图 2 和图 3.

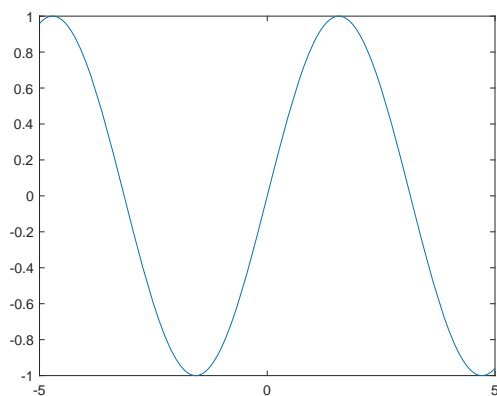


图 2 图像一的描述

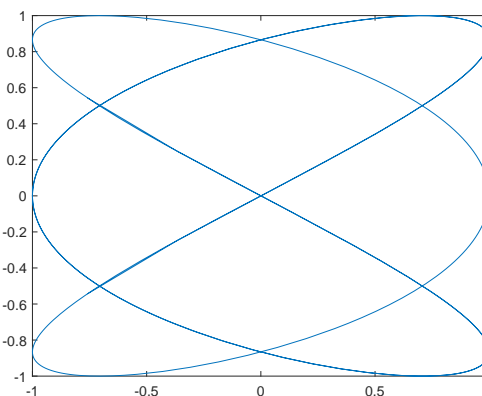


图 3 图像二的描述

代码高亮

这是 MATLAB 程序代码高亮环境.

MATLAB code

```
1 % Euler method for the ODE model
2 % u'(x)=x^2+x-u, x in [0,1]
3 % Initial condition: u(0)=0.
4 clear all;  clf
5 h=0.1;
6 x=0:h:1;
7 N=length(x)-1;
8 u(1)=0;
9 fun=@(t,u) t.^2+t-u;    % RHS
10 for n=1:N
11     u(n+1)=u(n)+h.*fun(x(n),u(n));
12 end
13 ue=-exp(-x)+x.^2-x+1; % exact solution
14 plot(x,ue,'b-',x,u,'r+', 'LineWidth',1)
15 legend('Exact','Numerical','location','North')
16 xlabel('x'), ylabel('u')
```

这是 Python 程序代码高亮环境.

Python code

```
1 #PythonDraw.py
2 import turtle as t
3 t.setup(650, 350, 200, 200)
4 t.penup()
5 t.fd(-250)
6 t.pendown()
7 t.pensize(25)
8 t.pencolor("purple color")
9 t.seth(-40)
10 for i in range(4):
11     t.circle(40, 80)
12     t.circle(-40, 80)
13 t.circle(40, 80/2)
14 t.fd(40)
15 t.circle(16, 180)
16 t.fd(40 * 2/3)
17 t.done()
```

参考文献

- [1] Tadmor E. A review of numerical methods for nonlinear partial differential equations[J]. Bull. Amer. Math. Soc., 2012, 49(4): 507-554.
- [2] 李荣华, 刘播. 微分方程数值解法[M]. 东南大学出版社, 1997.
- [3] Adams R A, Fournier J J F. Sobolev spaces[M]. Elsevier, 2003.
- [4] Trefethen L N, Weideman J A C. The exponentially convergent trapezoidal rule [J]. SIAM Rev., 2014, 56(3): 385-458.
- [5] Shen J. Efficient spectral-Galerkin method I. Direct solvers of second- and fourth-order equations using Legendre polynomials[J]. SIAM J. Sci. Comput., 1994, 15(6): 1489-1505.