



计算机与软件工程学院

上机报告

**（ 2021/2022 学年 第 1 学期 ）**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课程名称 | **数值计算（C++）** | | | | | |
| 课程代码 | **190901319** | | | | | |
| 上机时间 | 2021 | 年 | 12 | 月 | 22 | 日 |
| 指导单位 | 物联网工程系 | | | | | |
| 任课教师 | 李显勇 | | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 学生姓名 | 李子涵 | | |
| 学 号 | 3120190971401 | | |
| 成 绩 |  | 年级专业 | 计算机科学与技术 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **实验名称** | **【实验7】Euler方法和Runge-Kutta方法** | | | **实验地点** |  |
| **实验类型** | **验证** | **实验学时** | **2** | **实验日期** | 20211222 |
| **实验目的和要求**   1. 了解**求解常微分方程数值解的Euler方法和Runge-Kutta方法的**基本理论和算法； 2. 提交以下文档（用**学号+姓名**作为文件夹名，将所有内容放于该文件夹中）：   1）实验报告  2）程序代码 | | | | | |
| **实验环境(实验设备)**  windows XP及以上版本；PC；Matlab7.0及以上版本；其它高级语言 | | | | | |
| **实验原理及内容**  **一、实验原理**  根据**Euler方法和Runge-Kutta方法**相关知识和算法编程完成本实验  **二、实验内容**  已知微分方程         1. 用改进Euler方法求解微分方程； 2. 用四阶Runge-Kutta方法求解微分方程。   **三、实验过程（可以文字说明+运行结果截图）**  '''  文件名：西华大学数值计算C++实验报告——实验七  内容：欧拉公式与龙格公式  当前版本：1.0  完成作者：李子涵  学号：3120190971401  完成日期：2021.12.22  '''  import math  def dfx(x,y):      return y\*y\*math.cos(x)  def Euler(x0,y0,h,N):      """      x0,x1 旧值      y0y1 新值      h 步长      N 步数      yp yc 平均化的中间值      """      n=1      x1=0      while(1):          x1=x0+h          x1 = x0 + h          yp = y0 + h \* dfx(x0, y0)          yc = y0 + h \* dfx(x1, yp)          y1 = (yp + yc) / 2          print("x"+str(n)+"   "+"y"+str(n))          print(str(round(x1,2))+"   "+str(y1))          if n == N :              break          else:              x0 = x1              y0 = y1              n+=1  def Longe\_kutta(x0,y0,h,N):      """      x0,x1 旧值      y0y1 新值      h 步长      N 步数      k1 k2 k3 k4 平均化的中间值      """      n=1      x1=0      while(1):          x1=x0+h          K1 =dfx(x0,y0);          K2 = dfx(x0 + h / 2, y0 + h / 2\*K1);          K3 = dfx(x0 + h / 2, y0 + h / 2 \* K2);          K4 = dfx(x1, y0 + h \* K3);          y1 = y0 + h \* (K1 + 2\*K2 + 2\*K3 + K4) / 6;          print("x"+str(n)+"   "+"y"+str(n))          print(str(round(x1,2))+"   "+str(y1))          if n == N :              break          else:              x0 = x1              y0 = y1              n+=1  if \_\_name\_\_ == '\_\_main\_\_':      x0=0      y0=1      h=0.1      N=8      print("改进的欧拉公式:")      Euler(x0,y0,h,N)      print("龙格公式:")      Longe\_kutta(x0,y0,h,N)    实验结果：1：    2： | | | | | |

|  |
| --- |
| **四、实验小结**（包括问题和解决方法、心得体会、意见与建议等）  此实验较为简单，只需掌握算法的流程与循环结构即可。 |