# 牛鹏军 210711 21374389

## 理论分析

|  |
| --- |
| f5f72e83eaa593f24a4eb8829eeec1b |
| 669d2da3310f2139a6ca7c4a5015f66 |

利用matlab进行验证计算正确性：

|  |
| --- |
|  |

可见计算是正确的

## **算法设计**

用C语言分别写内置函数的黎曼积分解与近似解，比较其差异。

|  |
| --- |
| cf4d1e10d1c337f062690ccc9f5a5e2 |

## **编程实现**

|  |
| --- |
| #include <stdio.h>  #include <math.h>  double riemann\_integral(double (\*f)(double), double a, double b, int n) {  double dx = (b - a) / n; // 计算每个小矩形的宽度  double sum = 0.0;  double x;  for (int i = 0; i < n; i++) {  x = a + i \* dx; // 计算每个小矩形的左边界  sum += f(x) \* dx; // 高度乘以宽度，累加到总和中  }  return sum;  }  double lagrange\_integral(double (\*f)(double),double a, double b){  double c=(2\*a+b)/3;  double d=(a+2\*b)/3;  return (b-a)/8\*(f(a)+3\*f(c)+3\*f(d)+f(b));  }  // 示例函数：f(x) = x^2  double example\_function(double x) {  return x \* x;  }  int main() {  double a = 0.0; // 积分下限  double b = 1.0; // 积分上限  int n = 100000; // 小矩形的数量  double result\_riemann = riemann\_integral(example\_function, a, b, n);  double result\_lagrange = lagrange\_integral(example\_function, a, b);  printf("riemann\_result : %.4lf\n", result\_riemann);  printf("lagrange\_result: %.4lf\n", result\_lagrange);  return 0;  } |

## 测试分析

|  |
| --- |
|  |

用在0到1积分测试，精确值为1/3，可见用三次拉格朗日插值的方法积分精确度尚可。

## 结论

3次拉格朗日公式可推导出题目中的数值积分公式，用这种方法求近似解准确性还可以，比黎曼积分的求解步骤和时间显著缩短。