

**数值分析上机作业**

|  |  |
| --- | --- |
| 姓名： |  |
| 班级： |  |
| 学号： |  |
| 日期： |  |

|  |  |
| --- | --- |
| 题目IV | P195第37题：三次样条插值函数 |
| 题目V | P256第23题：重积分的计算 |

# 目录

[目录 2](#_Toc60334070)

[题目IV 1](#_Toc60334071)

[4.1 算法 1](#_Toc60334072)

[4.2 程序 1](#_Toc60334073)

[4.3 输出结果 2](#_Toc60334074)

[4.4 结果分析及感悟 3](#_Toc60334075)

[题目V 4](#_Toc60334076)

[5.1 算法 4](#_Toc60334077)

[5.2 程序 4](#_Toc60334078)

[5.3 输出结果 6](#_Toc60334079)

[5.4 结果分析及感悟 7](#_Toc60334080)

# 题目IV

1. 编制求第一型3次样条插值函数的通用程序；
2. 已知汽车门曲线型值点的数据如下：

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| i | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| xi | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| yi | 2.51 | 3.30 | 4.04 | 4.70 | 5.22 | 5.54 | 5.78 | 5.40 | 5.57 | 5.70 | 5.80 |

端点条件为=0.8，=0.2，用所编程序求车门的3次样条差值函数S（x），并打印出S（i+0.5），i=0,1······9。

## 4.1 算法

=

## 4.2 程序

clc;clear;

n=11;

xn=[0 1 2 3 4 5 6 7 8 9 10];

yn=[2.51 3.30 4.04 4.70 5.22 5.54 5.78 5.40 5.57 5.70 5.80];

dy0=0.8;

dyn=0.2;

d=zeros(n,1);

h=zeros(1,n-1);

f1=zeros(1,n-1);

f2=zeros(1,n-2);

for i=1:n-1

    h(i)=xn(i+1)-xn(i);                         %求一阶差值

    f1(i)=(yn(i+1)-yn(i))/h(i);

end

for i=2:n-1

    f2(i)=(f1(i)-f1(i-1))/(xn(i+1)-xn(i-1));    %求二阶差值

    d(i)=6\*f2(i);

end

d(1)=6\*(f1(1)-dy0)/h(1);

d(n)=6\*(dyn-f1(n-1))/h(n-1);

A=zeros(n);                                     %求M

u=zeros(1,n-2);

r=zeros(1,n-2);

for i=1:n-2

    u(i)=h(i)/(h(i)+h(i+1));

    r(i)=1-u(i);

end

A(1,2)=1;

A(n,n-1)=1;

for i=1:n

    A(i,i)=2;

end

for i=2:n-1

    A(i,i-1)=u(i-1);

    A(i,i+1)=r(i-1);

end

M=A\d;

digits(4);

syms x

for i=1:n-1                                     %求节点插值

    Sx(i)=collect(yn(i)+(f1(i)-(M(i)/3+M(i+1)/6)\*h(i))\*(x-xn(i))+M(i)/2\*(x-xn(i))^2+(M(i+1)-M(i))/(6\*h(i))\*(x-xn(i))^3);

    Sx(i)=vpa(Sx(i),4);

end

S=zeros(1,n-1);

for i=1:n-1

    x=xn(i)+0.5;

    S(i)=yn(i)+(f1(i)-(M(i)/3+M(i+1)/6)\*h(i))\*(x-xn(i))+M(i)/2\*(x-xn(i))^2+(M(i+1)-M(i))/(6\*h(i))\*(x-xn(i))^3;

end

disp('S(x)=');                                  %结果输出

for i=1:n-1

    fprintf('--------------------区间(%d,%d)---------------------\n',xn(i),xn(i+1));

    fprintf('    %s    \n',char(Sx(i)));

    disp('--------------------------------------------------');

end

disp('S(i+0.5)');

disp('     i    x(i+0.5)  S(i+0.5)')

for i=1:n-1

    fprintf('     %d    %.4f    %.4f\n',i,xn(i)+0.5,S(i))

end

## 4.3 输出结果

S(x)=

--------------------区间(0,1)---------------------

0.8\*x - 0.001486\*x^2 - 0.008514\*x^3 + 2.51

--------------------------------------------------

--------------------区间(1,2)---------------------

0.8122\*x - 0.01365\*x^2 - 0.004458\*x^3 + 2.506

--------------------------------------------------

--------------------区间(2,3)---------------------

0.8218\*x - 0.01848\*x^2 - 0.003654\*x^3 + 2.5

--------------------------------------------------

--------------------区间(3,4)---------------------

0.317\*x^2 - 0.1845\*x - 0.04092\*x^3 + 3.506

--------------------------------------------------

--------------------区间(4,5)---------------------

6.933\*x - 1.462\*x^2 + 0.1074\*x^3 - 5.984

--------------------------------------------------

--------------------区间(5,6)---------------------

4.175\*x^2 - 21.25\*x - 0.2685\*x^3 + 41.0

--------------------------------------------------

--------------------区间(6,7)---------------------

53.81\*x - 8.336\*x^2 + 0.4266\*x^3 - 109.1

--------------------------------------------------

--------------------区间(7,8)---------------------

6.247\*x^2 - 48.27\*x - 0.2679\*x^3 + 129.1

--------------------------------------------------

--------------------区间(8,9)---------------------

13.69\*x - 1.498\*x^2 + 0.05487\*x^3 - 36.18

--------------------------------------------------

--------------------区间(9,10)---------------------

14.55\*x - 1.593\*x^2 + 0.05838\*x^3 - 38.74

--------------------------------------------------

S(i+0.5)

i x(i+0.5) S(i+0.5)

1 0.5000 2.9086

2 1.5000 3.6784

3 2.5000 4.3815

4 3.5000 4.9882

5 4.5000 5.3833

6 5.5000 5.7237

7 6.5000 5.5944

8 7.5000 5.4299

9 8.5000 5.6598

10 9.5000 5.7323

## 4.4 结果分析及感悟

以上为各区间的三次样条插值函数及的结果。

三次样条插值是在每个小区间内分别建立样条函数，加上插值条件和连接条件，从而在整个区间上成立插值函数，大大提高了插值函数的精度。由于尽在节点处，函数值相等，故在别的数值处，存在误差。

通过上机编程，加深了对于三次样条插值函数的了解，同时也了解了其他多项式插值与函数最佳逼近，也提升了自己的编程能力。

# 题目V

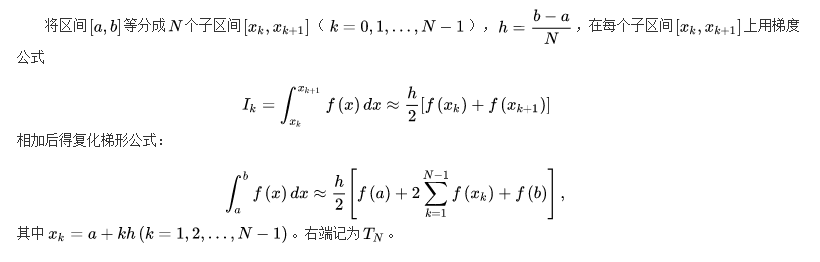
（1）给定积分，取初始步长和及精度，应用复化梯形公式，采用逐次二分步长的方法并应用外推思想编制计算的通用程序，计算至相邻两次近似值之差的绝对值不超过为止；

（2）用所编程序计算积分



取。

## 5.1 算法



## 5.2 程序

#include <iostream>

#include <math.h>

#define PI 3.1415926

#define error 0.5e-5

using namespace std;

double f(double x,double y)

{

    double out;

    out=tan(x\*x+y\*y);

    return out;

}

int main()

{

    double resultOld,temp,a=0,c=0;

    int i,j,m=1,n=1,time=0;

    double b=PI/3;

    double d=PI/6;

    double h=(b-a)/(2\*n);

    double k=(d-c)/(2\*m);

    double result=0;

    do

    {

        resultOld=result;

        result=0;

        result+=f(a,c);

        temp=0;

        for(i=1;i<n;i++)

            temp+=f(a+2\*i\*h,c);

        temp\*=2;

        result+=temp;

        temp=0;

        for(i=1;i<=n;i++)

            temp+=f(a+(2\*i-1)\*h,c);

        temp\*=4;

        result+=temp;

        result+=f(b,c);

        temp=0;

        for(j=1;j<m;j++)

            temp+=f(a,c+2\*j\*k);

        temp\*=2;

        result+=temp;

        temp=0;

        for(j=1;j<m;j++)

            for(i=1;i<n;i++)

                temp+=f(a+2\*i\*h,c+2\*j\*k);

        temp\*=4;

        result+=temp;

        temp=0;

        for(j=1;j<m;j++)

            for(i=1;i<=n;i++)

                temp+=f(a+(2\*i-1)\*h,c+2\*j\*k);

        temp\*=8;

        result+=temp;

        temp=0;

        for(j=1;j<m;j++)

            temp+=f(b,c+2\*j\*k);

        temp\*=2;

        result+=temp;

        temp=0;

        for(j=1;j<=m;j++)

            temp+=f(a,c+(2\*j-1)\*k);

        temp\*=4;

        result+=temp;

        temp=0;

        for(j=1;j<=m;j++)

            for(i=1;i<n;i++)

                temp+=f(a+2\*i\*h,c+(2\*j-1)\*k);

        temp\*=8;

        result+=temp;

        temp=0;

        for(j=1;j<=m;j++)

            for(i=1;i<=n;i++)

                temp+=f(a+(2\*i-1)\*h,c+(2\*j-1)\*k);

        temp\*=16;

        result+=temp;

        temp=0;

        for(j=1;j<=m;j++)

            temp+=f(b,c+(2\*j-1)\*k);

        temp\*=4;

        result+=temp;

        result+=f(a,d);

        temp=0;

        for(i=1;i<n;i++)

            temp+=f(a+2\*i\*h,d);

        temp\*=2;

        result+=temp;

        temp=0;

        for(i=1;i<=n;i++)

            temp+=f(a+(2\*i-1)\*h,d);

        temp\*=4;

        result+=temp;

        result+=f(b,d);

        result=result\*h\*k/9;

        m\*=2;

        n\*=2;

        h=(b-a)/(2\*n);

        k=(d-c)/(2\*m);

        cout<<"---------------------------------------------"<<"\n";

        cout<<"当前定积分结果是："<<result<<"\n";

        cout<<"当前相邻两次近似值之差的绝对值是："<<fabs(result-resultOld)<<"\n";

        time++;

    }while(fabs(result-resultOld)>error);

    cout<<"---------------------------------------------"<<"\n";

    cout<<"经过 "<<time<<" 次循环\n";

    cout<<"最终定积分近似值为:"<<result;

    cout<<"\n被划分成了"<<m/2<<"部分\n";

    return 0;

}

## 5.3 输出结果

---------------------------------------------

当前定积分结果是：0.396423

当前相邻两次近似值之差的绝对值是：0.396423

---------------------------------------------

当前定积分结果是：0.348623

当前相邻两次近似值之差的绝对值是：0.0477999

---------------------------------------------

当前定积分结果是：0.338354

当前相邻两次近似值之差的绝对值是：0.0102694

---------------------------------------------

当前定积分结果是：0.33672

当前相邻两次近似值之差的绝对值是：0.00163444

---------------------------------------------

当前定积分结果是：0.336537

当前相邻两次近似值之差的绝对值是：0.000182815

---------------------------------------------

当前定积分结果是：0.336522

当前相邻两次近似值之差的绝对值是：1.51861e-05

---------------------------------------------

当前定积分结果是：0.33652

当前相邻两次近似值之差的绝对值是：1.05544e-06

---------------------------------------------

经过 7 次循环

最终定积分近似值为:0.33652

被划分成了64部分

## 5.4 结果分析及感悟

初始时候在x，y方向上各二分一次，根据复化Simpson公式计算积分值，比较两次计算结果的差值，如果小于误差限，则已求得满足要求的结果，否则继续二分区间直到满足误差要求为止。最终经过7次二分，两次计算结果的绝对值差值为，所以得到最终定积分近似结果0.33652。