Problem Solving Through Programming in C

Tutorial Session 10

Prof. Anupam Basu
Dept. of Computer Science & Engg.
IIT Kharagpur

Siddhant Mohapatra
PMRF Scholar
IIT Madras

Interpolation

Q. A Lagrange polynomial passes through three data points as given below

x	10	15	20
$f(x)$	19.45	10.63	7.82

The polynomial is determined as $f(x) = L_0(x).(19.45) + L_1(x).(10.63) + L_2(x).(7.82)$

The value of $f(x)$ at $x = 12$ is

a) 16.20
b) 15.89
c) 15.20
d) 10.02

$$L_0(x)\Big|_{x=12} = \frac{12-15}{10-15} \times \frac{12-20}{10-20} = 0.48$$

$$L_1(x)\Big|_{x=12} = \frac{12-10}{15-10} \times \frac{12-20}{15-20} = 0.64$$

$$L_2(x)\Big|_{x=12} = \frac{12-10}{20-10} \times \frac{12-15}{20-15} = -0.12$$

at $x = 12$

$$f(x) = 0.48 \times 19.45 + 10.63 \times 0.64 - 0.12 \times 7.82 = 15.20.$$

Q. A Lagrange polynomial passes through three data points as given below

x	10	15	20
$f(x)$	3	5.2	6.8

The polynomial is determined as $f(x) = L_0(x).3 + L_1(x).(5.2) + L_2(x).(6.8)$

The value of $L_1(x)$ at $x = 18$ is

a) 0.64
b) 3.33
c) 2.67
d) 0.56

$$L_1(x) = \frac{x-10}{15-10} \times \frac{x-20}{15-20}$$

$x = 18$

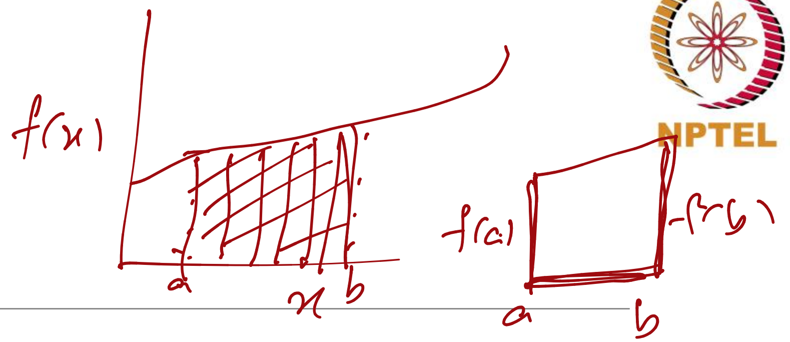
$$L_1(x) = \frac{18-10}{15-10} \times \frac{18-20}{15-20}$$

$$= 0.64$$

1 segment trapezoidal rule

$A = \frac{1}{2} \times (b-a) \times [f(a)+f(b)]$

$\int_a^b f(x)dx$



Interpolation and integration

Q. The value of $\int_0^{3.2} xe^x dx$ by using one segment trapezoidal rule is

a) 172.7
b) 125.6
c) 136.2
d) 142.8

$f(x) = xe^x$

$f(0) = 0$

$f(3.2) = 78.5$

$\frac{1}{2} \times (3.2-0) \times [0 + 78.5]$

$= 125.6$

$\text{Area} = \underbrace{\frac{1}{2} \times (3-0)}_{\text{height}} \times \underbrace{[f(0) + f(3)]}_{\text{sum of sides}}$

Q. The value of $\int_0^3 x^2 e^{2x} dx$ by using one segment trapezoidal rule is

a) 5446.3
b) 5336.2
c) 4986.5
d) 5278.4

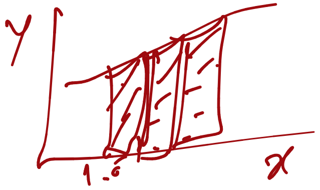
$f(x) = x^2 e^{2x}$

$f(0) = 0$

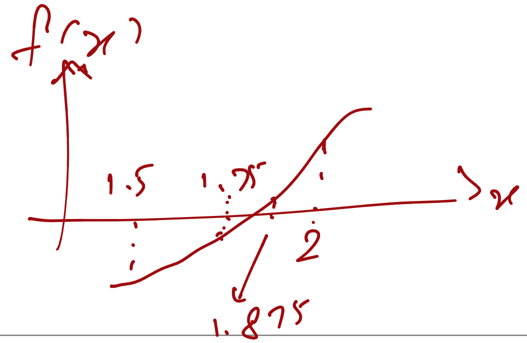
$f(3) = 3630.86$

$\frac{1}{2} \times (3-0) \times [0 + 3630.86]$

$= 5446.29 \approx 5446.3$



Interpolation and root finding



Q. What will be the area under the curve using the Trapezoidal Rule

x	1.4	1.6	1.8	2.0	2.2
y	4.3215	4.7428	5.5205	6.0525	6.8762

a) 4.3829 ✓
b) 5.4863
c) 6.3427
d) 3.2857

$$\text{Area} = \frac{1}{2} \times (1.6-1.4) \times (4.3215 + 4.7428)$$
$$+ \frac{1}{2} \times (1.8-1.6) \times (4.7428 + 5.5205)$$
$$+ \frac{1}{2} \times (2-1.8) \times (6.0525 + 5.5205)$$
$$+ \frac{1}{2} \times (2.2-2) \times (6.0525 + 6.8762)$$
$$= 4.3829$$

Q. Find the root of $x^4 - x - 10 = 0$ approximately upto 5 iterations using Bisection Method. Let a = 1.5 and b = 2.

a) 1.68
b) 1.92
c) 1.86 ✓
d) 1.66

$$f(x) = x^4 - x - 10$$
$$f(a) = -6.4375$$
$$f(b) = 4$$

1st iter

$$f(1.75) = -2.37$$

2nd iter

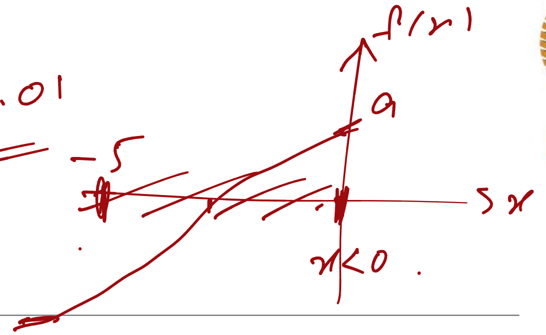
$$f\left(\frac{1.75+2}{2} = 1.875\right) = 0.4846$$

3rd iter

$$f\left(\frac{1.75+1.875}{2} = 1.8125\right) = -1.02$$

Root finding

$|x_{n+1} - x_n| < 0.01$



Q. The real root of the equation $5x - 2\cos x - 1 = 0$ (up to two decimal accuracy) is [You can use any method known to you. A range is given in output rather than single value to avoid approximation error]

a) 0.53 to 0.56 ✓
b) 0.45 to 0.47
c) 0.35 to 0.37
d) 0.41 to 0.43

$f(x) = 5x - 2\cos x - 1$

$f(x=0) = -3$

$f'(x) = 5 + 2\sin x$

$f'(x=0) = 5$

$x_0 = 0$

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = 0 - \frac{-3}{5} = 0.6$$

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} = 0.6 - \frac{0.349}{6.129} = 0.543 \approx 0.54$$

$$x_3 = x_2 - \frac{f(x_2)}{f'(x_2)} = 0.543 - \frac{2.675\times10^{-3}}{6.0334} = 0.5425 \approx 0.54$$

$|x_1 - x_0| = |0.6|$

$|x_2 - x_1| = |-0.057|$

$|x_3 - x_2| = 0.0005$

Q. Using Bisection method, negative root of $x^3 - 4x + 9 = 0$

a) -2.506
b) -2.706
c) -2.406
d) None of the above

$f(x) = x^3 - 4x + 9$

$f(0) = 9$

$f(-5) = -96$

$f(-2.5) =$

Trapezoidal rule (integration)

$f(x) = \ln x$

Q. The value of $\int_{2.5}^{4} \ln x \, dx$ calculated using the Trapezoidal rule with five subintervals is (* range is given in output rather than single value to avoid approximation error)

a) 1.45 to 1.47
b) 1.74 to 1.76
c) 1.54 to 1.56
d) 1.63 to 1.65

$f(2.5) = 0.9163$
$f(2.8) = 1.0296$
$f(3.1) = 1.1314$
$f(3.4) = 1.2238$
$f(3.7) = 1.3083$
$f(4) = 1.3863$

I II III IV V

2.5 2.8 3.1 3.4 3.7 4

$2.5 + \frac{1}{5}(4-2.5)$

$= 2.8$

$2.5 + \frac{2}{5}(4-2.5) = 3.1$

$\frac{1}{2} \times (2.8-2.5) \times \begin{bmatrix} 0.9163 + \\ 1.0296 \end{bmatrix} + \frac{1}{2} \times (3.1-2.8) \times \begin{bmatrix} 1.0296 + \\ 1.1314 \end{bmatrix} + \frac{1}{2} \times (3.4-3.1) \times \begin{bmatrix} 1.1314 + \\ 1.2238 \end{bmatrix} + \frac{1}{2} \times (3.7-3.4) \times \begin{bmatrix} 1.2238 + \\ 1.3083 \end{bmatrix} + \frac{1}{2} \times (4-3.7) \times \begin{bmatrix} 1.3083 + \\ 1.3863 \end{bmatrix}$

RK-4 method

To solve the ordinary differential equation

$$10\frac{dy}{dx} + x^2e^x = y\cos(x) + x\sin(y)\,, y(0) = 5$$

using Runge-Kutta 4th order method, the equation is re-written as

a) $\frac{dy}{dx} = y\cos(x) + x\sin(y)\,, y(0) = 5$

b) $\frac{dy}{dx} = \frac{1}{10}(y\cos(x) + x\sin(y))\,, y(0) = 5$

c) $\frac{dy}{dx} = \frac{1}{10}(y\cos(x) + x\sin(y) - x^2e^x), y(0) = 5$

d) $\frac{dy}{dx} = x\sin(y) - x^2e^x, y(0) = 5$

$$10\frac{dy}{dx} = y\cos x + x\sin y - x^2e^x$$

$$\underbrace{\frac{dy}{dx} = \frac{1}{10}\left(y\cos x + x\sin y - x^2e^x\right)}$$

$$\frac{dy}{dx} = f(x,y)$$ — rewrite & then use RK4.

RK-4 method

Q. Given $4\frac{dy}{dx}+x^2=y^3$, y(0.5)=2, and using a step size of $h=0.2$, Find the value of $y(0.7)$ using Runge-Kutta 4th order method is

a) 2.8634
b) 2.5546
c) 2.1865
d) 1.9856

$$\frac{dy}{dx} = (y^3 - x^2)\frac{1}{4}$$

$$= f(x,y)$$

$$f(x,y) = \frac{1}{4}(y^3 - x^2)$$

$$y(x=0.5) = 2$$

$$k_1 = f(x,y)$$

$$k_2 = f\left(x+\frac{h}{2}, y+\frac{hk_1}{2}\right)$$

$$k_3 = f\left(x+\frac{h}{2}, y+\frac{hk_2}{2}\right)$$

$$k_4 = f(x+h, y+hk_3)$$

$$k_1 = f(x=0.5, y=2) = 1.9375$$

$$k_2 = f\left(0.5+\frac{0.2}{2}, 2+\frac{0.2}{2}\times 1.9375\right) = f(0.6, 2.19375) = 2.5494$$

$$k_3 = f(0.5+0.1, 2+0.1\times 2.5494) = f(0.6, 2.25494) = 2.7764$$

$$k_4 = f(0.5+0.2, 2+0.2\times 2.7764) = f(0.7, 2.55528) = 4.0486$$

$$y(x+h) = y(x) + \frac{h}{6}(k_1+2k_2+2k_3+k_4)$$

$$y(0.7) = 2 + \frac{0.2}{6}(1.9375+2\times 2.5494+2\times 2.7764+4.0486) = 2.5546$$

Structures

$a+ib$

Q. Write a C program to add two complex numbers using a function.

```
#include<stdio.h>
struct complex{        tag
    float a;
    float b;   } members
}comp;   user-defined datatype

comp   addComplex (comp   num1, comp   num2)
{
    complex  sum;
    sum.a = num1.a + num2.a;
    sum.b = num1.b + num2.b;
    return sum;
}
int main( )
{
    complex num1, num2;
    num1.a = 5.5; num2.a = -2.5;
    num1.b = -3.1; num2.b = 10.2;

complex sum = addComplex(num1, num2);
```

specify diff

what time should a variable exist

Storage specifiers

- automatic/local
- external/global
- static
- register

} Scope

faster than local variables

Static variables: example

Q.

```
#include<stdio.h>
int fun()
{
    static int count = 0;
    count++;
    return count;
}

int main()
{
    printf("%d ", fun());
    printf("%d ", fun());
    return 0;
}
```

static var.

1st fun()
count = 0
count ++ => 1
return

static

2nd fun()
count = 1
count ++ => 2
returned

1 2

Q.

```
#include<stdio.h>
int fun()
{
    int count = 0;
    count++;
    return count;
}

int main()
{
    printf("%d ", fun());
    printf("%d ", fun());
    return 0;
}
```

local var.

1 1

local - variable is destroyed once function call is over.

accessible to all function

global - var is destroyed at end of code execution.

static]
it is local to fn where it is defined

0, 1, 1, 2, 3, 5, 8, 13, 21 . . .

Recursion

Write a C program to print the Fibonacci sequence (upto 10 terms) using recursion.

```c
#include<stdio.h>
int fibonacci_[redacted](int prev, int new);
int term = 10;
int main()
{
    int prev = 0, new = 1;
    printf("%d, %d, ", prev, new);
    term -= 2;
    fibonacci(prev, new);
    return 0;
}
```

→ global variable

term = 8 – global variable is modified.

```c
int fibonacci_[redacted](int prev, int new)
{
    static int i = 0;
    int temp;
    if (i == term)    return 0;
    else {
        temp = prev + new;
        prev = new;
        new = temp;
        printf("%d  ", new);
        i++;
        fibonacci(prev, new);
    }
    return 0;
}
```

static var.

recursion stops ←

0, 1, 1, 2, 3, 5, 8, 13, 21, 34 ⇒ 10 terms of Fibonacci seq.

i: 0, 1, 2, 3, 4, 5, 6, 7, 8