Name : Deepankar Sharma
Course: BCA
University Roll No: 2092014
Student Id : 20041299
Semester: 3
Date: December 29, 2021

Practical 14: Integration

Objective: To integrate a given function over given limit using :

1. Trapezoidal Rule
2. Simpson's 1/3 Rule
3. Simpson's 3/8 Rule

2. Algorithm:

1. Start
2. Define the function f(x)
3. Input ll, ul corresponding to lower limit and upper limits.
4. Input n, number of intervals
5. Calculate h=(ul-ll)/n
6. Make Table:

```
Define mat[n][2]
Set z= ll, lower limit

for ( i=0 to n)
{
   mat[i][0]=(z)
   mat[i][1]=f(z)
   z+=h
}
```

7. Print Table.

8. Trapezoidal Rule:

```
sum= mat[0][1]+ mat[n][1]
   for (int i = 1; i < n; i++)
   {
      sum+= 2*mat[i][1]
   }
   sum*=(h/2)
```

9. Simpson's 1/3 Rule:

```
if(n%2 !=0){
      Print("Can't apply Simpson's 1/3 Rule !!!!\n\n");
       return -1;
```

```
        }

    else{
        sum = mat[0][1] + mat[n][1]
        sumEven=0.0, sumOdd=0.0

        for (int i = 1 to n-1)
        {
            if (i%2==0)
            {
                sumEven+= 2*mat[i][1]
            }
            else
                sumOdd += 4 * mat[i][1]
        }
    sum+=sumEven+sumOdd
    sum*=(h/3);
    }
```

10. Simpson's 3/8 Rule:

```
    if (n % 3 != 0)
    {
        Print("Can't apply Simpson's 3/8 Rule !!!!\n\n");
        return -1;
    }

    else
    {
        sum = mat[0][1] + mat[n][1]
        sum3X = 0.0, sumOthers = 0.0
        for (int i = 1 to n-1)
        {

            if (i % 3 == 0)
            {

                sum3X += 2 * mat[i][1]
            }
            else
             sumOthers += 3 * mat[i][1]; // 3*y[i]
        }
    sum += sum3X + sumOthers;
    sum *= ((3*h) / 8);
    }
```

11. Do{
        Input choice
        Switch(choice){
            case 1: trapezoidalRule(h, n);
                break;
            case 2: simpson1By3(h, n);
                break;
            case 3: simpson3By8(h, n);
                break;

            default:
                EXIT

    }While(true)
12. Print Sum.
13. Stop.

Code:

```
#include<iostream>
#define f(x) 1/(1+ (x*x))

float ll; // lower limit
float ul; // upper limit
int n;    // number of intervals
static float mat[10][2] = {0.0};

float trapezoidalRule(float h, int n){
   float sum= mat[0][1]+ mat[n][1]; // y[0]+y[n]
   for (int i = 1; i < n; i++)
   {
      /* code */
      sum+= 2*mat[i][1]; // 2*y[i]
   }
   sum*=(h/2);
   printf("The integration of the given function over the limit %f to %f using Trapezoidal
Rule is %f\n\n", ll, ul, sum);
   return sum;
}

float simpson1By3(float h, int n){
   if(n%2 !=0){
```

```
        printf("Can't apply Simpson's 1/3 Rule !!!!\n\n");
        return -1;
    }

    else{
        float sum = mat[0][1] + mat[n][1]; // y[0]+y[n]
        float sumEven=0.0, sumOdd=0.0;

        for (int i = 1; i < n; i++)
        {
            /* code */
            if (i%2==0)
            {
                /* code */
                sumEven+= 2*mat[i][1]; // 2*y[i]
            }
            else
                sumOdd += 4 * mat[i][1]; // 4*y[i]
        }
        sum+=sumEven+sumOdd;
        sum*=(h/3);
        printf("The integration of the given function over the limit %f to %f using Simpson's
1/3 Rule is %f\n\n", ll, ul, sum);
        return sum;
    }

}

float simpson3By8(float h, int n){
    if (n % 3 != 0)
    {
        printf("Can't apply Simpson's 3/8 Rule !!!!\n\n");
        return -1;
    }

    else
    {
        float sum = mat[0][1] + mat[n][1]; // y[0]+y[n]
        float sum3X = 0.0, sumOthers = 0.0; // sum3X---> multiples of 3, sumOthers---->
others

        for (int i = 1; i < n; i++)
        {
            /* code */
```

```
            if (i % 3 == 0)
            {
                /* code */
                sum3X += 2 * mat[i][1]; // 2*y[i]
            }
            else
                sumOthers += 3 * mat[i][1]; // 3*y[i]
        }
        sum += sum3X + sumOthers;
        sum *= ((3*h) / 8);
        printf("The integration of the given function over the limit %f to %f using Simpson's
3/8 Rule is %f\n\n", ll, ul, sum);
        return sum;
    }
}

using namespace std;
int main(){
    cout<<"Enter the lower limit: ";
    cin>>ll;
    cout<<"Enter the upper limit: ";
    cin>>ul;
    cout<<"Enter the number of intervals: ";
    cin>>n;

    float h= (ul-ll)/n;
    float z= ll;

    for (int i = 0; i <=n; i++)
    {
        /* code */
        mat[i][0]=(z); // x[i]
        mat[i][1]=f(z); // y[i]
        z+=h;
    }
    cout<<endl<<endl;

    // printing the table
    cout << "x: " ;
    for (int i = 0; i < n; i++)
    {
        /* code */
        printf("%.6f\t", mat[i][0]);
    }
```

```
    cout<<endl;
    cout << "y: " ;
    for (int i = 0; i < n; i++)
    {
       /* code */
       printf("%.6f\t", mat[i][1]);
       // cout << mat[i][1] << "\t";
    }

    cout<<endl<<endl;


    do
    {
       /* code */
       int choice=0;
       cout<<"(1). Trapezoidal Rule"<<endl;
       cout<<"(2). Simpson's 1/3 Rule"<<endl;
       cout<<"(3). Simpson's 3/8 Rule"<<endl;
       cout<<"(4). Exit"<<endl;
       cout<<"Using which rule would you like to integrate?  : ";
       scanf(" %d", &choice);

       switch (choice)
       {
       case 1: trapezoidalRule(h, n);
          /* code */
          break;
       case 2: simpson1By3(h, n);
          /* code */
          break;
       case 3: simpson3By8(h, n);
          /* code */
          break;

       default:
          cout<<"EXIT"<<endl;
          goto x;
       }

    } while (1);

    x:return 0;
}
```

Output:

```
Windows PowerShell
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS E:\03 Semester\CBNST\Unit 04> cd "e:\03 Semester\CBNST\Unit 04\" ; if ($?) { g++
14_Integration.cpp -o 14_Integration } ; if ($?) { .\14_Integration }
Enter the lower limit: 0
Enter the upper limit: 1
Enter the number of intervals: 6


x: 0.000000    0.166667    0.333333    0.500000    0.666667    0.833333
y: 1.000000    0.972973    0.900000    0.800000    0.692308    0.590164

(1). Trapezoidal Rule
(2). Simpson's 1/3 Rule
(3). Simpson's 3/8 Rule
(4). Exit
Using which rule would you like to integrate?  : 1
The integration of the given function over the limit 0.000000 to 1.000000 using Trapezoidal Rule is
0.784241

(1). Trapezoidal Rule
(2). Simpson's 1/3 Rule
(3). Simpson's 3/8 Rule
(4). Exit
Using which rule would you like to integrate?  : 2
The integration of the given function over the limit 0.000000 to 1.000000 using Simpson's 1/3 Rule is
0.785398

(1). Trapezoidal Rule
(2). Simpson's 1/3 Rule
(3). Simpson's 3/8 Rule
(4). Exit
Using which rule would you like to integrate?  : 3
The integration of the given function over the limit 0.000000 to 1.000000 using Simpson's 3/8 Rule is
0.785396

(1). Trapezoidal Rule
(2). Simpson's 1/3 Rule
(3). Simpson's 3/8 Rule
(4). Exit
Using which rule would you like to integrate?  : 4
EXIT
```