Department of Computer Science

Gujarat University

Certificate

Roll No: 36 *Seat No:* ______

This is to certify that Mr./Ms. PREKSHA K. SHETH *student of MCA Semester – II has duly completed his/her term work for the semester ending in June 2020, in the subject of* COMPUTER ORIENTED AND NUMERICAL METHODS *towards partial fulfillment of his/her Degree of Masters in Computer Applications.*

Date of Submission *Internal Faculty*

1st - JULY - 2020

Head of Department

Department Of Computer Science
Rollwala Computer Centre
Gujarat University

MCA - II

Subject: - Comm ~~Numerical~~ [illegible]

Name: - Preksha K. Sheth

Roll No.: - 36 **Exam Seat No.: -** ________

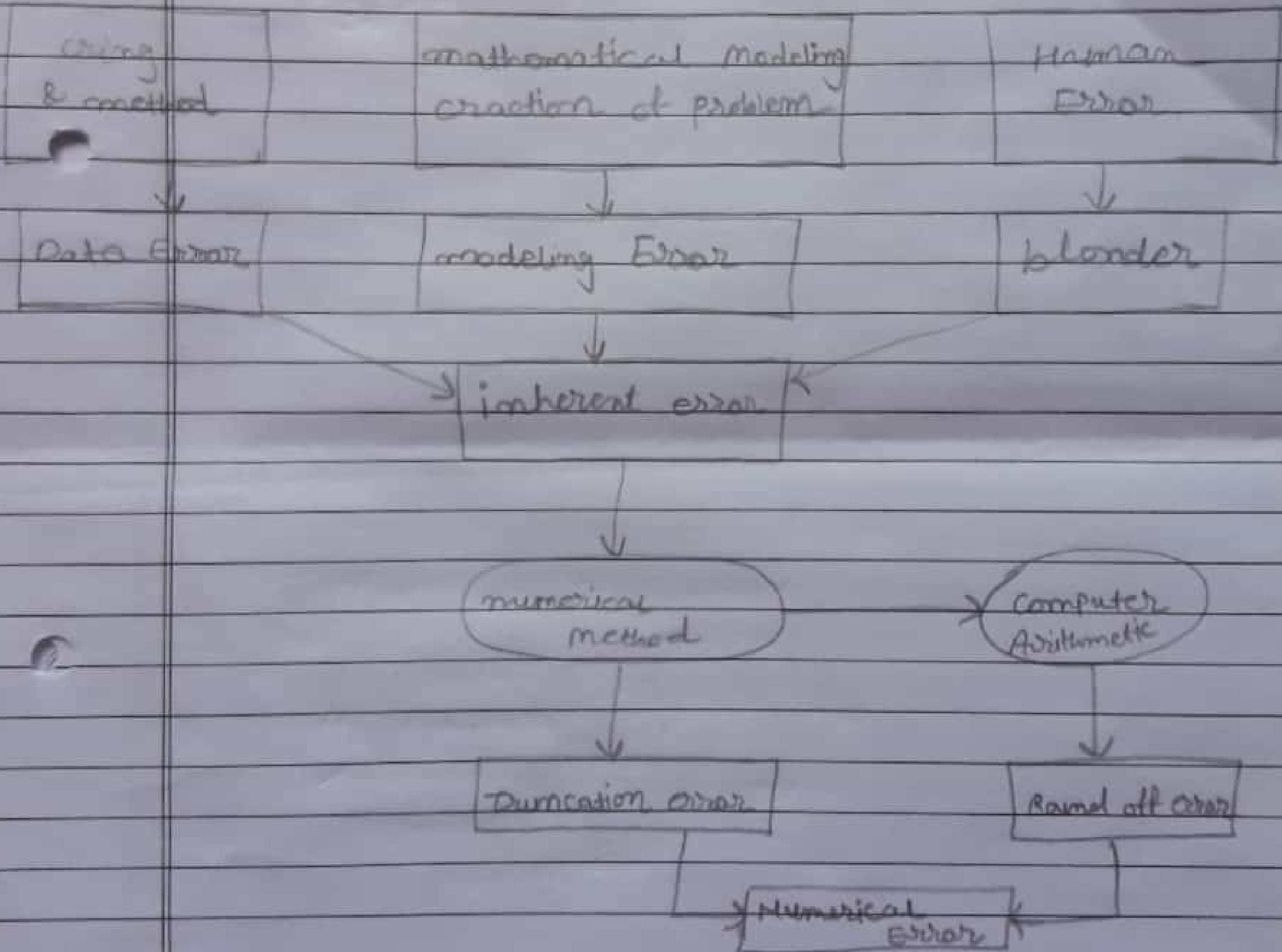
Sr. No.	Contents	Pg. No	Date	Signature
1	types of Errors and sources.		10/2/2020	[illegible]
2.	methods to find roots of Equation.		10/2/2020	
3	Numerical Integration		1/7/2020	
4.	Open type integration formula & Gauss Quaderature formula		1/7/2020	
5	Solution of ordinary numerical methods		1/7/2020	
6	ordinary differiential equation - II		1/7/2020	
7	Numerical Differentiation		1/7/2020	

Scanned with CamScanner

Assignment-1

Q.1 Explain the diffrent types of errors that occur during computation.

* Sources of errors:



Truncation error :

→ Truncation Error is an error in implementation of numerical approximation method occuring due to truncating a process of infinite number of steps to finite number of steps.

i) limitting infinite series to finite number of terms
ii) limiting infinite number of iterations to finite number of iteration.

* Round off error:

→ Round off errors occur due to finite precision in a computer a number may not always have a finite representation, eg.

$\frac{1}{3} = 0.3333...$
$\sqrt{2} = 1.4142135623.$

* Error propogation: -

Every arithmetic computation in digital arithmetic leads to compunding of rounding off errors as follows:

$$E_{a+b} = E_a + E_b + r_a$$

Rounding off error increase with increase in computations. The higher the number of steps the more shall be the rounding error.

Assignment 2

Q.1 Explain the Bisection method graphically.

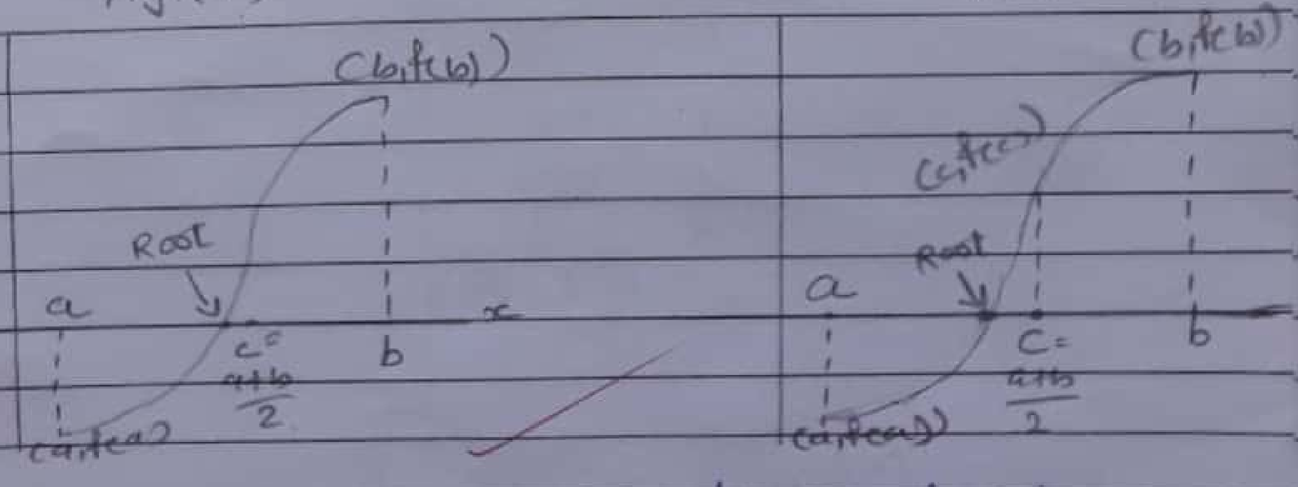
→ Bisection method begins with two initial guesses a & b. Such that $f(a) \cdot f(b) < 0$. This ensures that at least one root lies between a & b.

→ Now $c = (a+b)/2$. So the interval [a, b] is divided into exactly 2 equal parts [a, c] & [c, b].

→ This can be visualized from the following graph. c is our estimate of the root.

→ Now accordingly to f(c), c will take the role of a or b to ensure that root always lies within a & b.

fig.(1) fig(2)



Root lies b/w a & b

$f(c) > 0$, $f(b) \cdot f(c) > 0$

∴ c takes the Role of b

∴ c is now new b.

→ we continue the process, till our interval becomes sufficiently small.

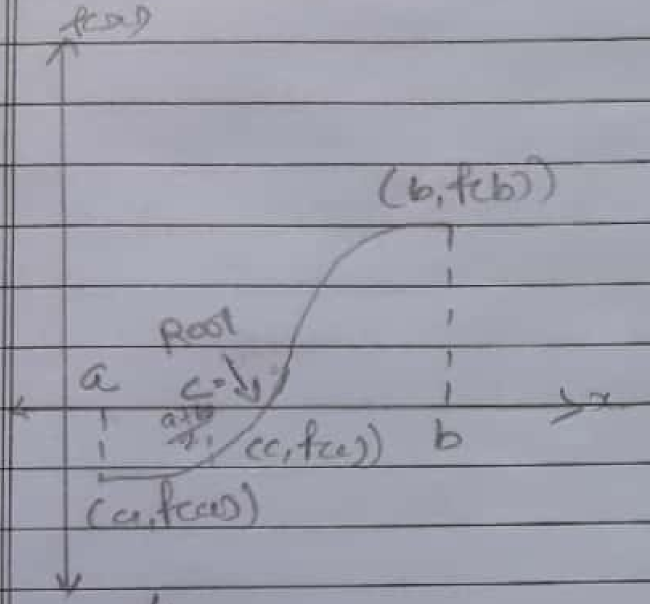


fig (3)

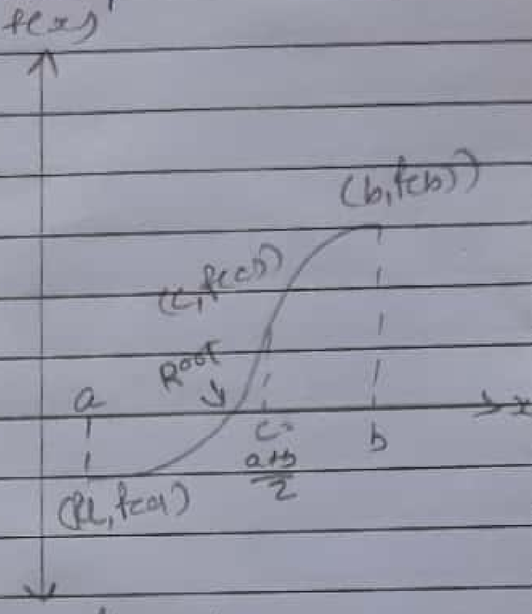
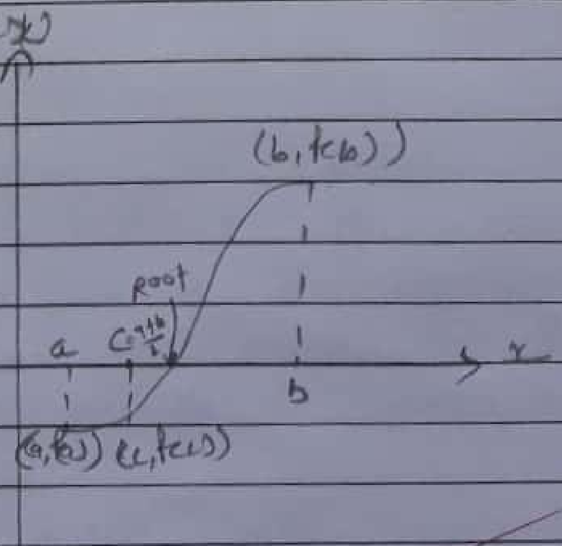


fig (4)

→ $f(c) < 0$, $f(a) \cdot f(c) > 0$
∴ c takes role of a.
c is near a.

→ $f(c) > 0$, $f(b) \cdot f(c) > 0$
∴ c takes the role of b.
c is near b.



→ As number of iterations increases the interval becomes smaller & we get our root as c.

Page No. 5
Date

Q2 Explain the Advantages & Disadvantages of Bisection method.

Ans Advantages:

→ It is very simple.
→ One initial a, b are known, number of iterations needed to be performed to achieve desired accuracy can be predetermined.
→ It is reliable method
→ It guarantees convergence.
→ function needs to be only continuous.
→ only one function evaluation per iteration.
→ Calculation for making guess of root for the next iteration is very easy. simply $c = \frac{a+b}{2}$.

* Disadvantages:

→ Not self starting, to begin with, require two initial guesses at which function must of opposite sign. Thus one has to evaluate function at subinterval points to get such a & b.
→ Slowest method.
→ It doesnot take into account the nature of function to make next guess of the root.

Teacher's Signature :

Q.3 Find root of the following using Bisection method:

1) $f(x) = x^3 - x - 1 = 0$.

no.	a	f(a)	b	f(b)	$c = \frac{a+b}{2}$	f(c)
1	1	−1	2	5	1.5	0.8750
2	1	−1	1.5	0.8750	1.25	−0.2969
3	1.25	−0.2969	1.5	0.8750	1.3750	0.2246
4	1.25	−0.2969	1.3750	0.2246	1.3125	−0.0515
5	1.3125	−0.0515	1.3750	0.2246	1.3438	0.0826
6	1.3125	−0.0575	1.3438	0.0826	1.3282	0.0146
7	1.3125	−0.0515	1.3282	0.0146	1.3204	−0.0186
8	1.3204	−0.0186	1.3282	0.0146	1.3243	−0.0018
9	1.3243	−0.0018	1.3282	0.0146	1.3263	0.0065
10	1.3243	−0.0018	1.3263	0.0065	1.3253	0.0025
11	1.3243	−0.0018	1.3253	0.0025	1.3248	0.0003
12	1.3243	−0.0018	1.3248	0.0003	1.3246	−0.0007
13	1.3246	−0.0007	1.3247	0.0003	1.3247	−0.0001
14	1.3247	−0.0001	1.3248	0.0003	1.3248	0.0003

$\therefore$ Root = 1.3248

b) $f(x) = xe^{x} = 1$

$\therefore f(x) = xe^{x} - 1 = 0$.

no	a	f(a)	b	f(b)	c	f(c)
1	0	-1	1	1.7183	0.5	-0.1756
2	0.5	-0.1756	1	1.7183	0.75	0.5878
3	0.5	-0.1756	0.75	0.5878	0.625	0.1677
4	0.5	-0.1756	0.625	0.1677	0.5625	-0.0128
5	0.5625	-0.0128	0.625	0.1677	0.5938	0.0751
6	0.5625	-0.0128	0.5938	0.0751	0.5782	0.0307
7	0.5625	-0.0128	0.5782	0.0307	0.5704	0.0089
8	0.5625	-0.0728	0.5704	0.0089	0.5665	-0.0019
9	0.5665	-0.0019	0.5704	0.0089	0.5685	0.0036
10	0.5665	-0.0019	0.5685	0.0036	0.5675	0.0010
11	0.5665	-0.0019	0.5675	0.0010	0.5670	-0.0004
12	0.5670	-0.0004	0.5675	0.0010	0.5673	0.0003
13	0.5670	-0.0004	0.5673	0.0003	0.5672	0.0000

$\therefore$ Root = 0.5672.

Q4 How many iterations do you need in Bisection method to get root if you start with $a = 1$ & $b = 2$ & the tolerance is 10^{-4}?

→ In Bisection method,
if e is the tolerance limit then the approximation number of the itrations needed to be performed is given

by the formula:

$$n \geq [\log(b-a) - \log \epsilon] / \log 2.$$

where a, b are starting point & n is the iteration number.

→ we can derive this as follows:
Initially interval is length b-a, Root lies between a, b. Initial estimate of the Root is midpoint of interval [a, b]. Therefore maximum error is less than or equal to half of the length of the interval.

i.e. $|E| \leq b-a/2$

→ At each step, the length of the interval becomes half of the previous one
∴ so at the iterations, $|E| \leq (b-a)/2^2$.

→ if $|E| < \epsilon$ then we require $(b-a/2^n) \leq \epsilon$ at n^{th} step

→ Taking log on both sides,
$$\log(b-a) - n\log 2 \leq \log \epsilon$$

$$\therefore n \geq [\log(b-a) - \log \epsilon] / \log 2.$$

→ Here, $a=1$ & $b=2$ & tolerance is $\epsilon = 10^{-4}$.

So minimum iterations required is

$$n \geq [\log(2-1) - \log 10^{-4}] / \log 2$$
$$= [\log 1 - (-4)\log 10] / \log 2$$
$$= [0-(-4)\times 1] / 0.3010$$
$$= 13.2890$$

∴ minimum iterations needed is 14.

Q.5 Explain the False position or Regula Falsi method graphically.

→ False position method is also a closed method like bisection. so at the beginning of iterative process two points needed so that $f(a) \cdot f(b) < 0$ to ensure that at least one root lies between a & b.

→ we can get our estimate of the root c by $c = \dfrac{a f(b) - b f(a)}{f(b) - f(a)}$

→ now according the f(c), c takes place of a or b.

→ Here is the graphically explaination,

→ case 1 : c takes sale of a as f(c)<0
& f(a) . f(c) >0

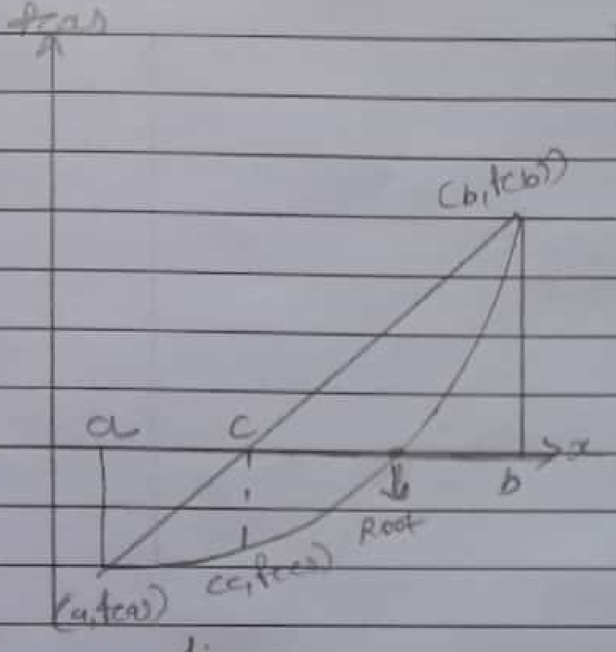


fig (1)

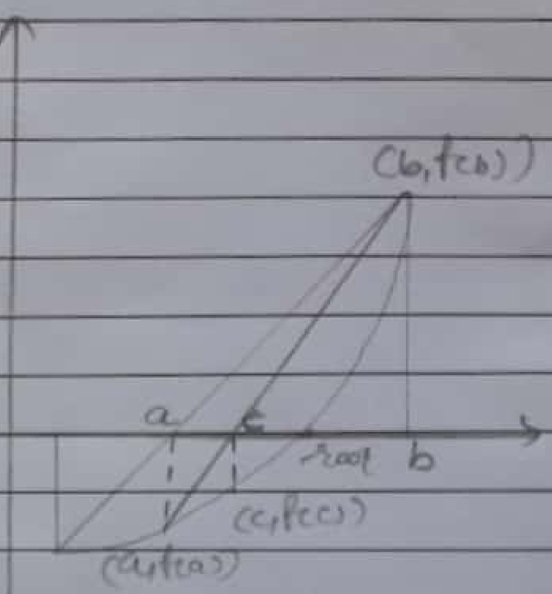


fig (2)

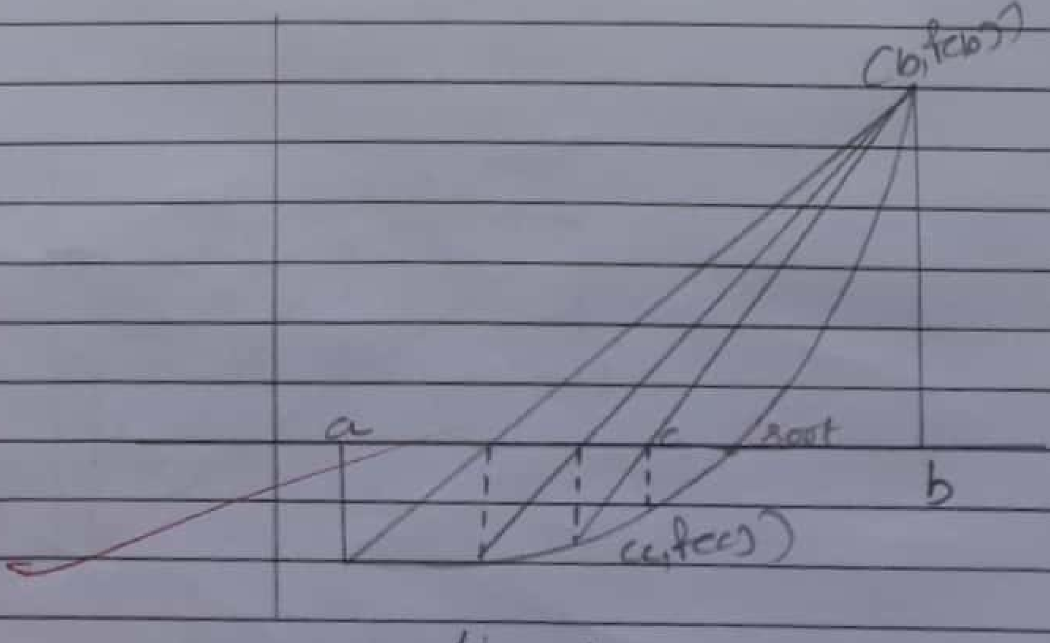
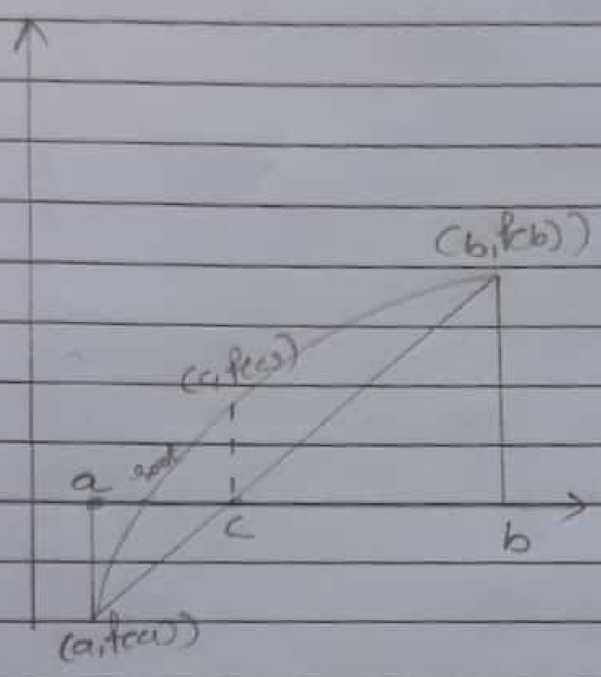


fig (3)

Case 2: $f(c) > 0$ & $f(b) \cdot f(c) > 0$

$\therefore$ c takes role of b



(fig 4)

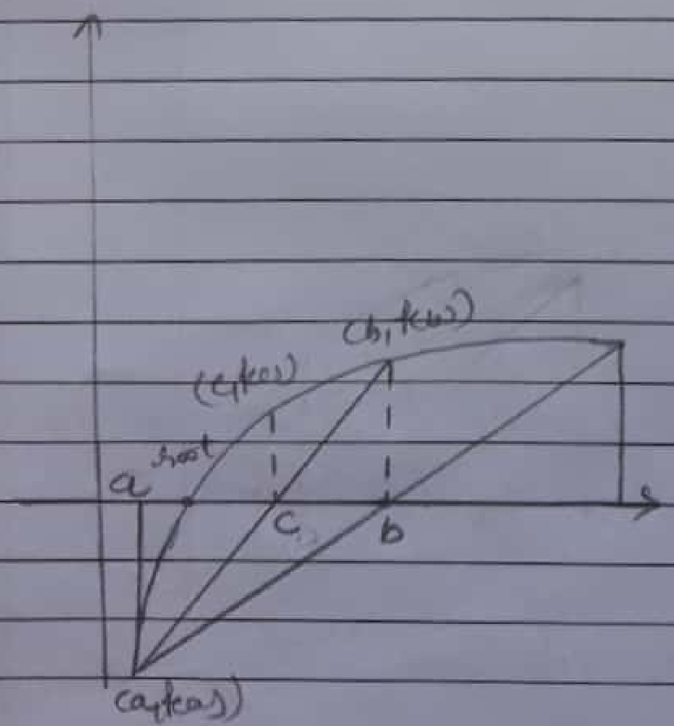


fig. (5)

Case 2:

$f(c) > 0$ & $f(b) \cdot f(c) > 0$ $\therefore$ c takes role of b.

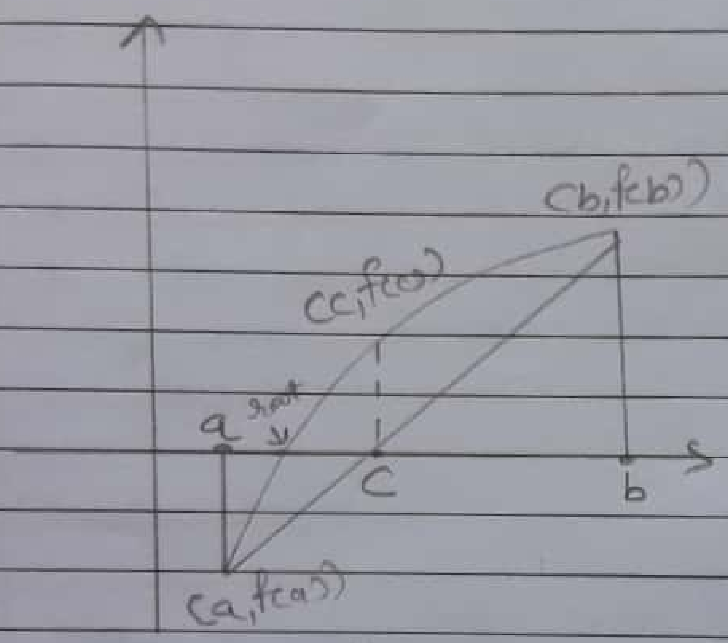


fig. (4)

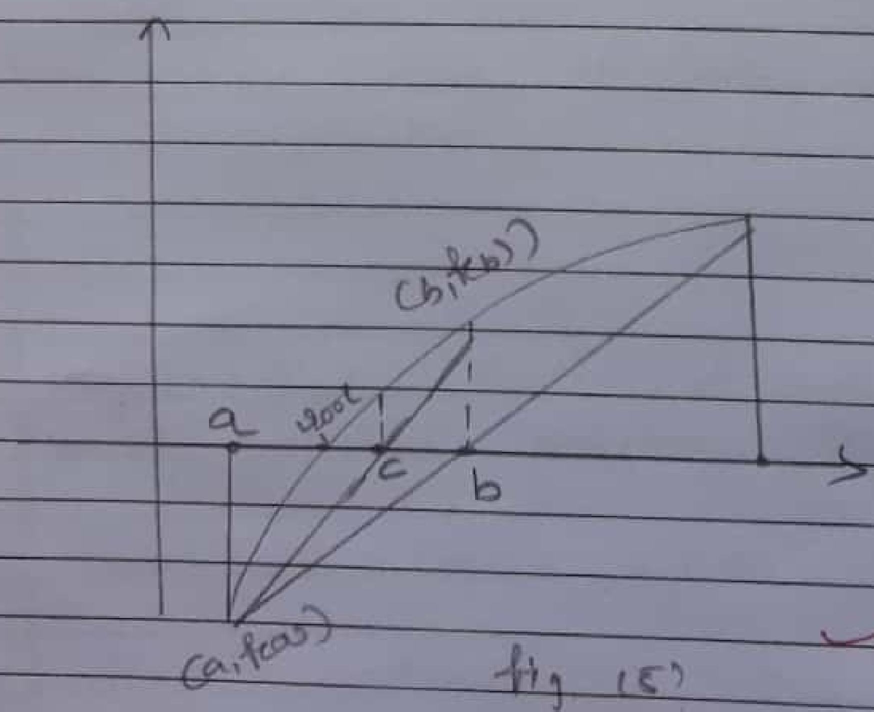


fig. (5)

Q.6 Explain the advantages & disadvantages of false position method.

Ans. Advantages:

→ It is faster than bisection method.
→ It is very simple to implement
→ It guarantees convergence.
→ only one function evaluation per iteration is required

Disadvantages:

→ Not self starting. we need to set two initial guesses a & b such that $f(a) \cdot f(b) < 0$.
→ Though, faster than bisection method, still regarded as slow.
→ In rare case, it may become slower than Bisection.

Q.7 Find the root of the following using false position method.

1) $f(x) = x \log_{10} x = 1.2$

$f(x) = x \log_{10} x - 1.2 = 0.$

no	a	f(a)	b	f(b)	c $\frac{bf(a)-af(b)}{f(a)-f(b)}$	f(c)
1	2	-0.5979	3	0.2314	2.7210	-0.0171
2	2.7210	-0.0171	3	0.2314	2.7402	-0.0004
3	2.7402	-0.0004	3	0.2314	2.7406	0.0000

Root = 2.7406

b) $f(x) = x^3+x-1 = 0.$

no	a	f(a)	b	f(b)	c	f(c)
1	0	-1	1	1	0.5	-0.3750
2	0.5	-0.3750	1	1	0.6364	-0.1059
3	0.6364	-0.1059	1	1	0.6712	-0.0264
4	0.6712	-0.0264	1	1	0.6797	-0.0064
5	0.6797	-0.0064	1	1	0.6817	-0.0014
6	0.6817	-0.0014	1	1	0.6821	-0.0004
7	0.6821	-0.0004	1	1	0.6822	-0.0002
8	0.6822	-0.0002	1	1	0.6823	-0.0002
9	0.6823	-0.0002	1	1	0.6824	0.0001

∴ Root = 0.6824

Teacher's Signature :

Q.8 Explain the advantages & disadvantages of Secant method.

→ Advantages

→ open method, no constraints of end points of interval to contain the root.
→ If converges, it converges quite fast, super linear convergence of order 1.618.
→ Requires only one function evaluation per iteration.

→ Disadvantages:

→ No more guarantee of convergence.
→ may diverge, if initial guess are not choosen cautiously.

Q.9 find the root of following using secant method.

a) $f(x) = 3x - \cos x - 1$

Iter.	~~a~~ x_{n-1}	~~f(a)~~ $f(x_{n-1})$	~~b~~ x_n	f(b)

no.	x_{n-1}	$f(x_{n-1})$	x_n	$f(x_n)$	x_{n+1} $\frac{x_n f(x_{n-1}) - x_{n-1} f(x_n)}{f(x_{n-1}) - f(x_n)}$	$f(x_{n+1})$
1	0	−2	1	1.4597	0.5781	−0.1033
2	1	1.4597	0.5781	−0.1093	0.6060	−0.0040
3	0.5781	−0.1033	0.6060	−0.0040	0.6071	0.0000

b) $f(x) = xe^x = 1$ $\qquad f(0) = -1$

$= xe^x - 1 = 0.$ $\qquad f(1) = 1.7183$

no	x_{n-1}	$f(x_{n-1})$	x_n	$f(x_n)$	x_{n+1}	$f(x_{n+1})$
1	0	−1	1	1.7183	0.3679	−0.4685
2	1	1.7183	0.3679	−0.4685	0.5033	−0.1674
3	0.3679	−0.4685	0.5033	−0.1674	0.5786	0.0319
4	0.5033	−0.1674	0.5786	0.0319	0.5665	−0.0016
5	0.5786	0.0319	0.5665	−0.0016	0.5671	−0.0002
6	0.5665	−0.0016	0.5671	−0.0002	0.5672	0.0001
7	0.5671	−0.0002	0.5672	0.0001	0.5672	0.0001

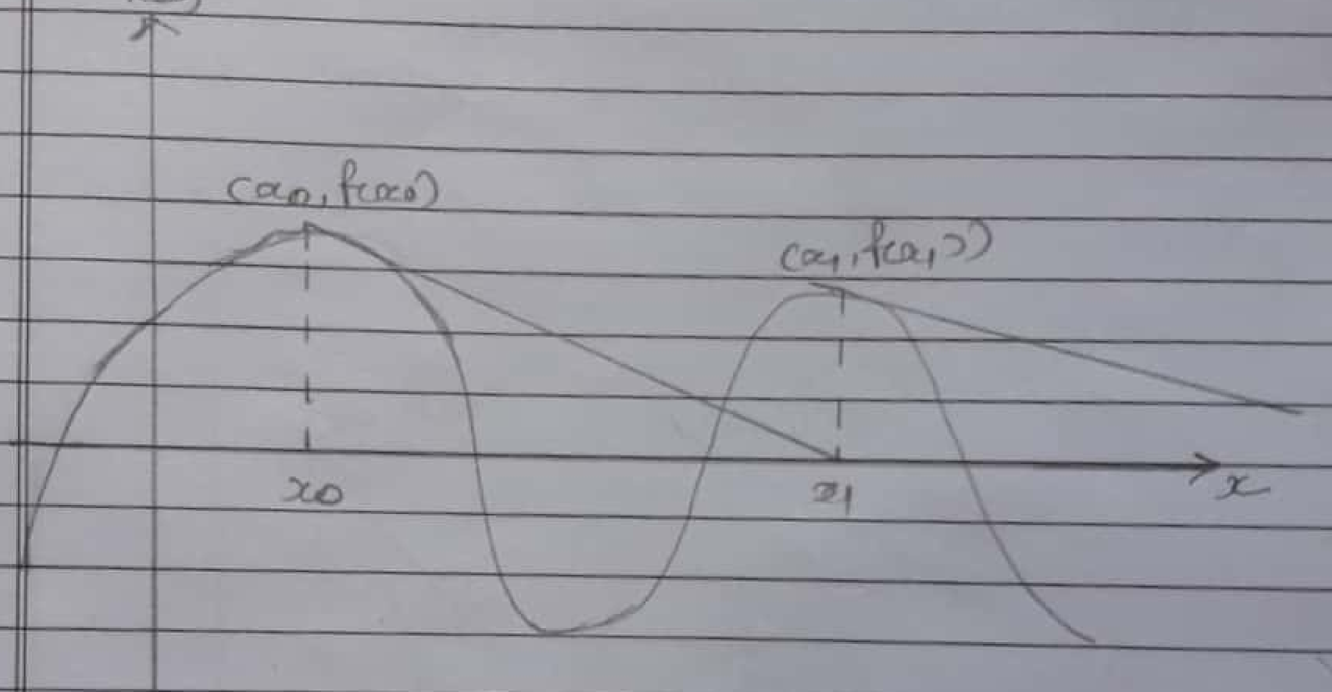
Root = 0.5672

Q.10 Explain graphically the conditions in which the Newton Raphson method fails to converge

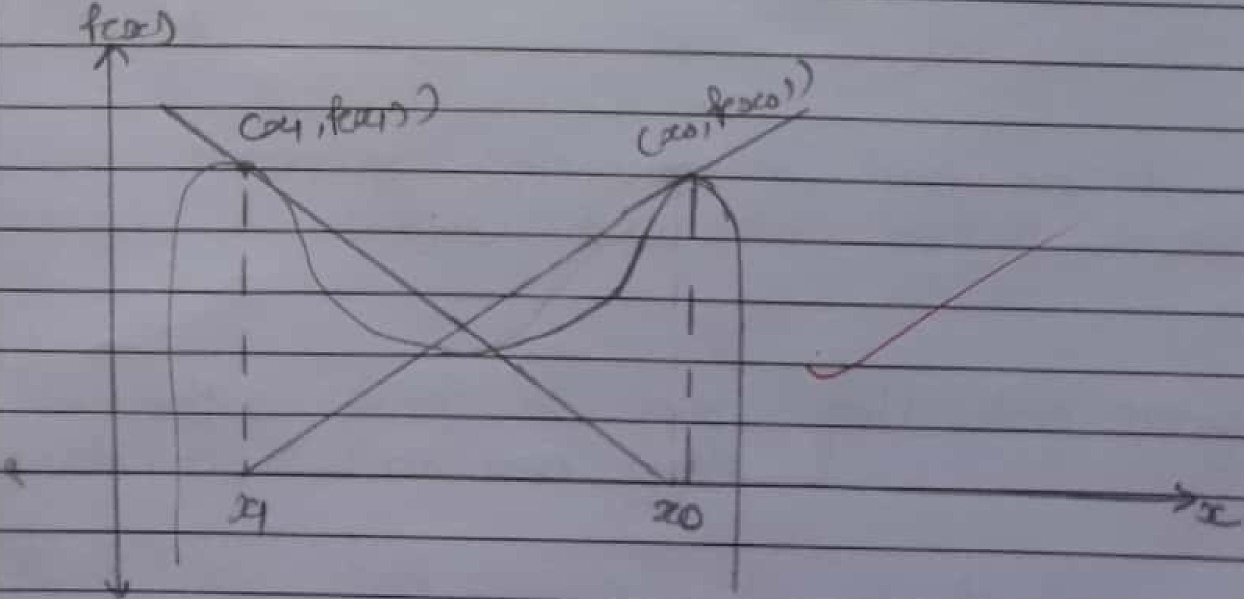
Ans In the following cases the newton Raphson method falls to converge.

Case 1:

→ Tangent is almost parallel ($f'(1) \simeq 0$), leading to halting of the process.



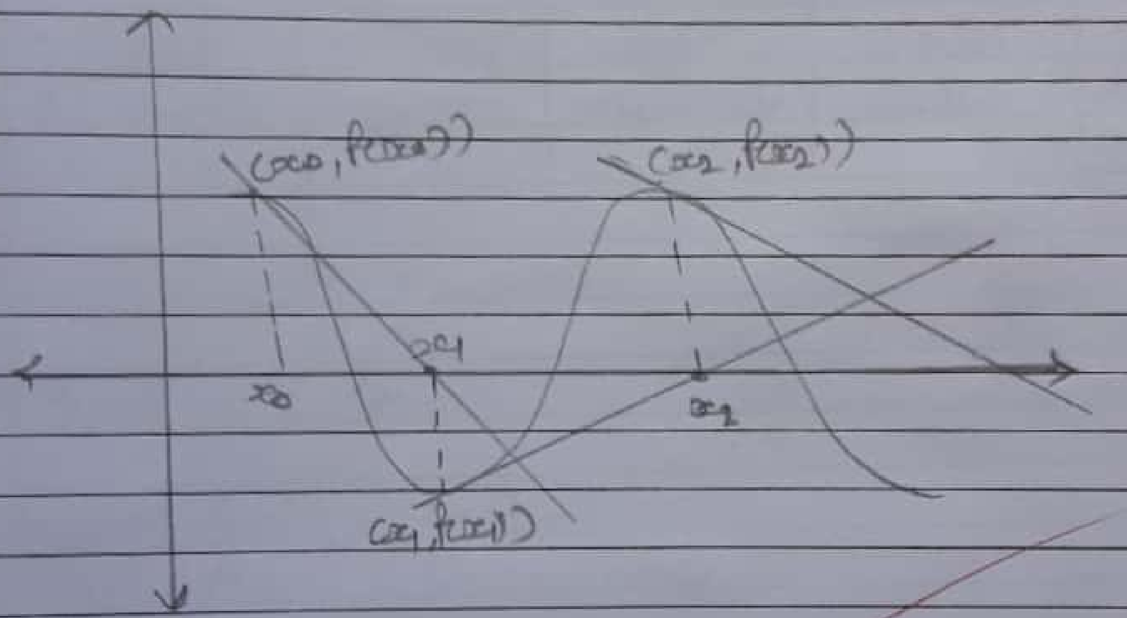
* Case 2: oscillations:



→ If we look at the above fig tangent to the curve $y = f(x)$ at $x = x_0$ meets x-axis.

at x_1, & tangent to the curve at $x=x_1$ meets x-axis at x_0, Thus the approximations shall keep on oscillating between two values x_0 & x_1, setting an infinite loop, without leading anywhere.

* Case 3: Divergence.



→ Here successive approximations keep on becoming larger & larger thus the method diverges.

Q.11 find the root of the following using Newton Raphson method.

1) find an approximation to $\sqrt{5}$ to four decimal places.

Page No.
Date

Ans.

Here,

$f(x) = \sqrt{5} = x$

$f(x) = 5 - x^2 = 0$

$f'(x) = -2x$

no.	x_n	$f(x_n)$	$f'(x_n)$	$x_{n+1} = x_n - \frac{f(x)}{f'(x_n)}$	$f(x_{n+1})$
1	2	1	-4	2.25	-0.0625
2	2.25	-0.0625	-4.5	2.2361	-0.0002
3	2.2361	-0.0002	-4.4722	2.2361	-0.0002

$\therefore \sqrt{5} = 2.2361$

b) $f(x) = x - 2\sin x$.

here $f(x) = x - 2\sin x = 0$

$f'(x) = 1 - 2\cos x = 0$.

no.	x_n	$f(x_n)$	$f'(x_n)$	x_{n+1}	$f(x_{n+1})$
1	2	0.1814	1.8323	1.9010	0.0090
2	1.9010	0.0090	1.6485	1.8955	0.0000

$\therefore$ Root $= \underline{1.8955}$

Teacher's Signature :

Q13 Find the Root of following using fixed point.

1) $f(x) = 3x - \cos x - 1$

here, $x = (\cos x + 1)/3 = g(x)$

no.	x	$g(x)$
1	0	0.6667
2	0.6667	0.5953
3	0.5953	0.6093
4	0.6093	0.6067
5	0.6067	0.6072
6	0.6072	0.6071
7	0.6071	0.6071

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

here $x = e^{-x} = g(x)$

no.	x	$g(x)$
1	0	1
2	1	0.3679
3	0.3679	0.6922
4	0.6922	0.5005
5	0.5005	0.6062
6	0.6062	0.5454
7	0.5454	0.5796
8	0.5796	0.5601
9	0.5601	0.5711
10	0.5711	0.5649
11	0.5649	0.5684

Teacher's Signature :

12	0.5664	0.5664
13	0.5664	0.5676.
14	0.5676	0.5669.
15	0.5669	0.5673.
16	0.5673	0.5671
17	0.5671	0.5672.
18	0.5672	0.5671

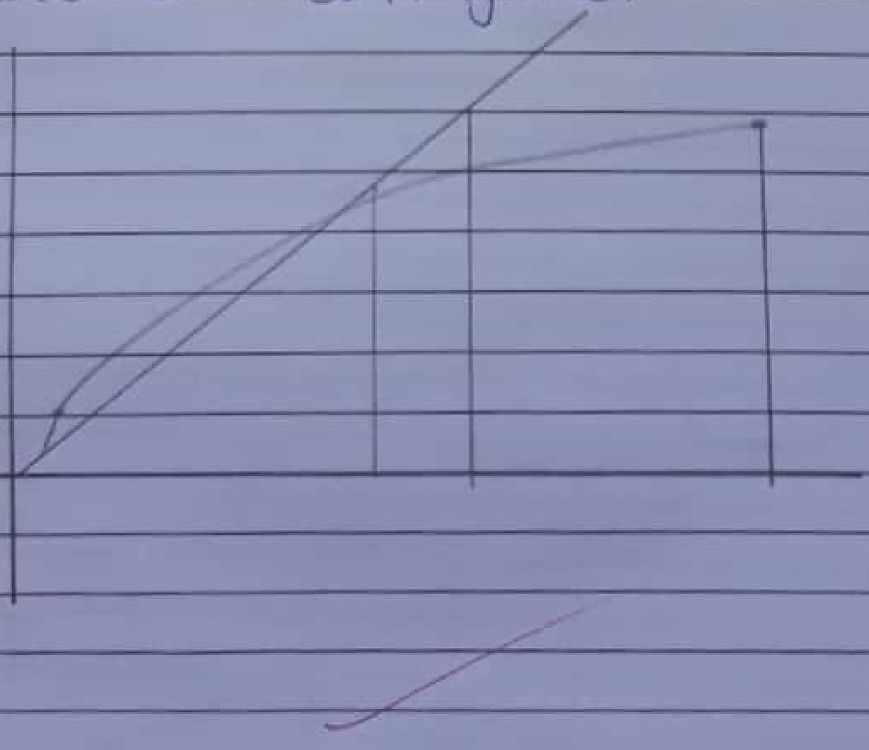
∴ Root = 0.5671.

Q.14 Explain graphically convergence & divergence of the fixed point method.

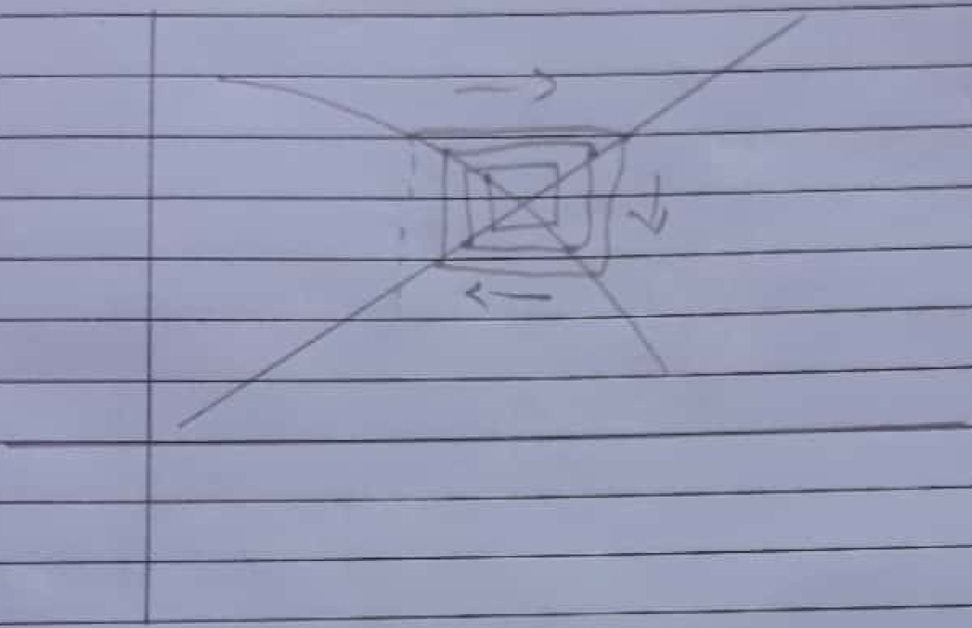
→ There are four main type of convergence & divergence of the fixed point method.

A→ let g(x) equal the derivative of the function g evaluated at the fixed point x.

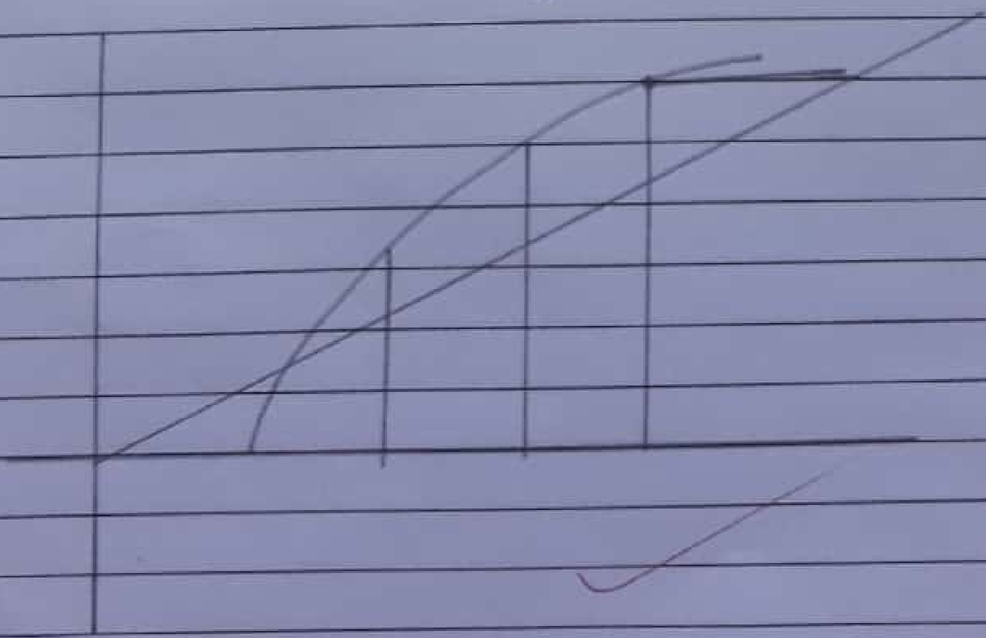
1) monotonic convergence:



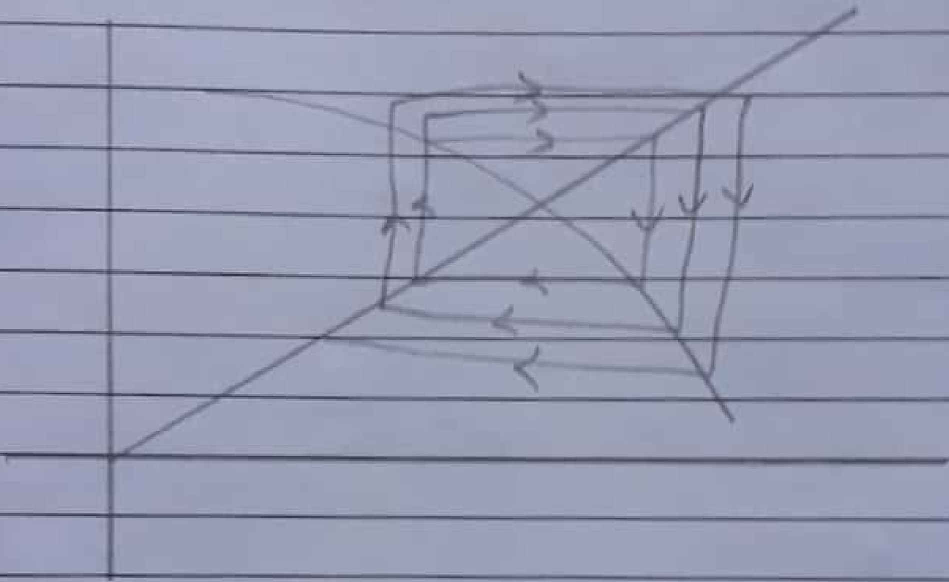
2) Oscillating Convergence:



3) monotonic divergence:



* Oscillating divergence:



Assignment . 2.

Q1 What is the need of numerical Integration? Explain different situation where such need arises.

→ In Intigration, whenever anti derivative is known, indefinite integral is known. The difficulty arises in Computing definite integral in many situation coming to different reasons.

Situation I :

→ For evaluating $\int_a^b f(x)dx$, anti-derivative of f exists but cannot be represented of in form of standard functions. for example, Consider

i) $\int_0^1 e^{x^2} dx$ & ii) $\int_1^2 \frac{dx}{\log x}$

one may say, anti-derivative of e^{x^2} is $\int_0^x e^{t^2} dt$, but is of no use to us. Both these dted integrals have precise numerical values for all practical purposes. anti-derivative does not exist in above two case as integral expressions cannot be wed to calculate the values F and thus the value of definite integral cannot be determined.

→ many such examples are encountered in scientific, mathematical & statistical and engineering problems at
1) The error function

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$, used in calculating

error bounds in many computations is of great importance in statistics.

2) whereas $\int_2^x \frac{dt}{\ln(t)}$ is logarithm function,

which gives approximately no. of primes less than or equal to x

3) In electric field theory, it is proved that the magnetic field induced by a current flowing in a circular loop of wire has intensity,

$$H(x) = \frac{4Ir}{r^2-x^2} \int_0^{\pi/2} \left[1-\left(x/r\right)^2 \sin^2\theta\right]^{\frac{1}{2}} d\theta$$

→ where I is the current, r is the radius of the loop, x is the distance from center to the point, where the magnetic intensity is being computed $(0 \leq x \leq r)$. If r & I are given, the integral occuring in the evalation is caued elliptic integral & cannot be expressed in term of standard function.

→ All three integrals are frequently occurring integrals & we shall learn numerical method called methods of numerical integration. Using these methods, we would be able to evaluate these integrals to as many decimal places as desired.

Situation 2 :

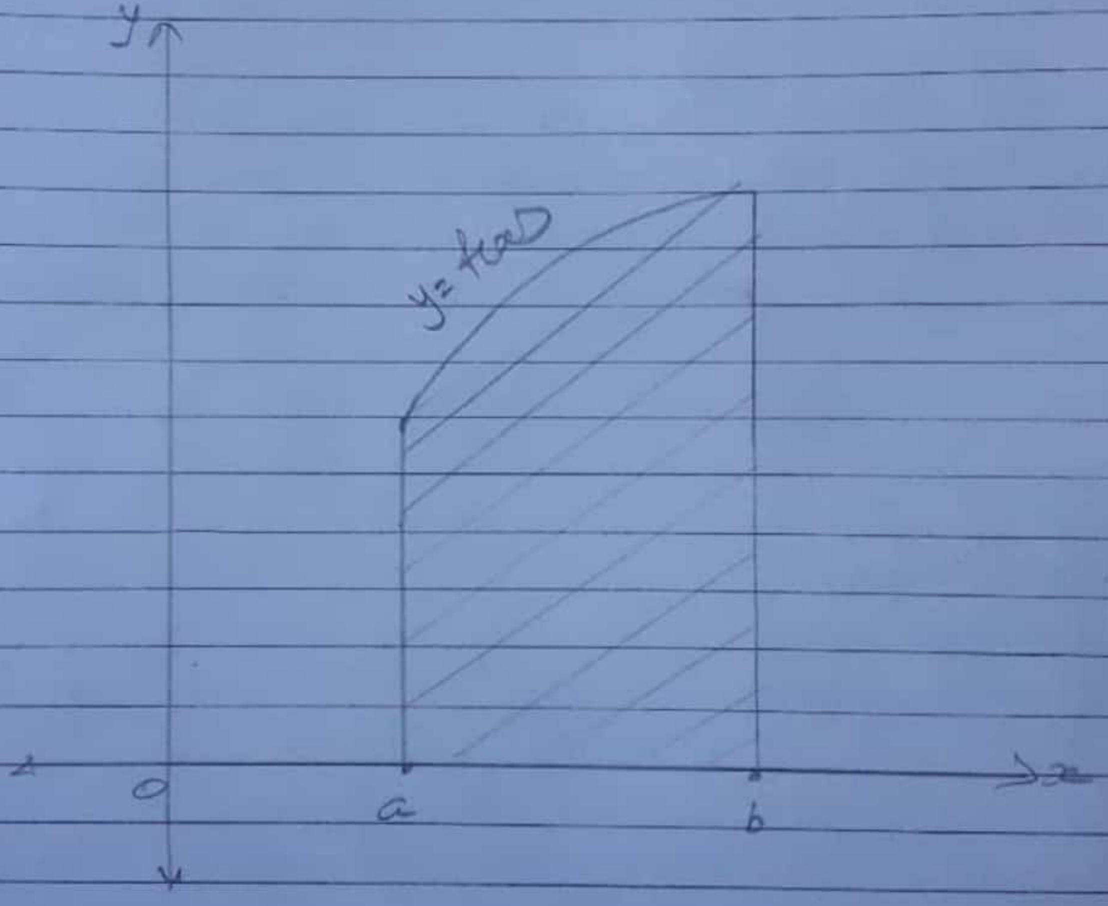
→ Anti-derivative may exist, known but may be complicated expression to compute. In these cases rather than evaluate antiderivative, it's preferred to use numerical integration formula to evalute the integral. A typical situation would be where antiderivative expression is in the form of an infinite series or infinite product & its evaluation requires numerical techniques

Situation 3 :

→ Expression of function to be integrated is not available. Functions are defined with the help of discrete data. for example, speed of an object may have been measured at different time intervals & distance needs to be computed or integrated may have been obtained by sampling, so values are available at certain points only.

Q.2 Give geometrical interpretation of $\int_a^b f(x)dx$, with $f(x) \geq 0$ & $\leq x \leq b$.

→ The definite integral $\int_a^b f(x)dx$ of a non-negative integrand $f(x)$ on a Closed interval $[a,b]$ is area under the graph of f.



Q.3 Differentiate between:
(a) Newton cotes Integration formulas & Gauss Quadrative formulas.
(b) Closed type & open type formulas.

→ (a) Here x_i are called abscissas & w_i are called weights.:

→ In first case x_i's are fixed, that is known & w_i's are computed by fitting function to the data $(x_i, f(x_i))$ & integrating the resulting fitted function. The x_i's points lies within the interval $[a,b]$ of integration.

→ When the fitting function is a polynomial, the integration formulas so obtained are called Newton Cotes integration formulas.

→ On the other hand, if x_i's & w_i's both are assumed to be unknown & they are computed, the formulas so obtained are called Gauss Quadrature formulas.

Integration Formulas

x_i's fixed w_i's unknown Newton Cotes Integration formulas	x_i's unknown w_i's unknown Gauss Quadrature formulas

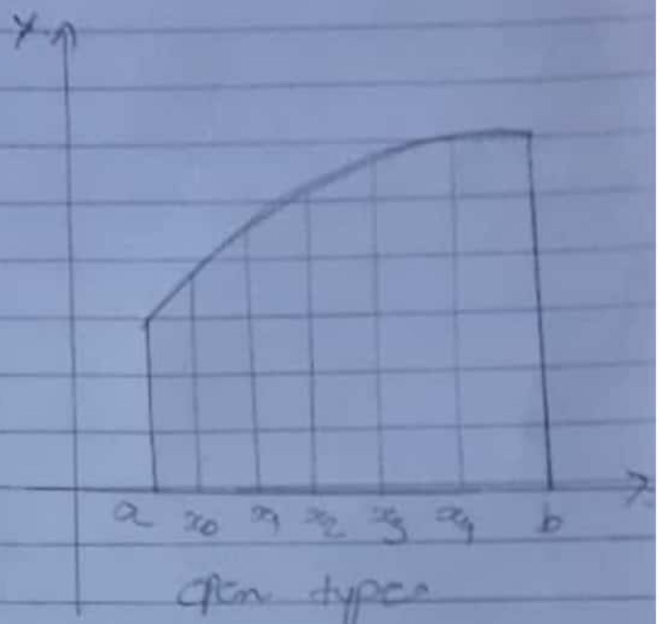
b) Closed type & open type formulas:

→ If $x_0 = a$ & $x_n = b$, i.e. $a = x_0 \le x_1 \le \ldots \le x_{n-1} \le x_n = b$ then integration formulas is called Closed type Integration formulas.

→ If $a < x_0 \le x_1 \le x_2 \le \ldots \le x_{n-1} \le x_n < b$ then the integration formulas is called open type integration formulas.

→ The fig. makes it clear

closed type.

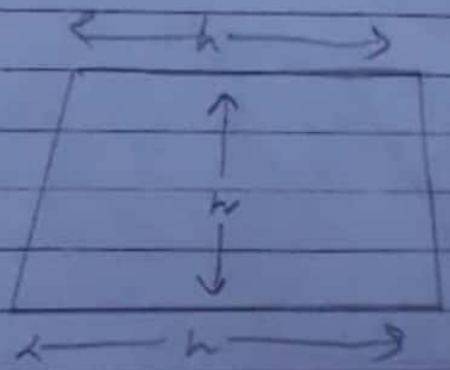


open types

Q4 Explain trapezoidal rule for estimating $\int_a^b f(x)\,dx$. Explain it graphically also.

→ By trapezoidal rule, $\int_a^b f(x)\,dx = \frac{\Delta x}{2}[f(a)+f(b)]$

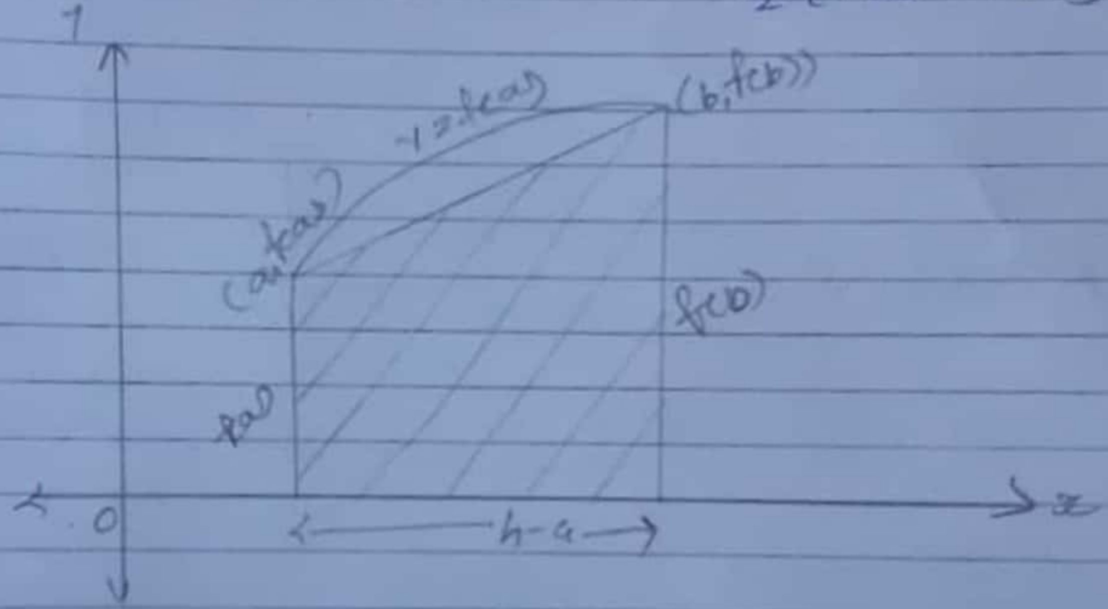
→ This rule is called trapezoidal rule as it gives area of trapezium formed from joining the points $(a, f(a))$ & $(b, f(b))$ by a straight line & the vertical lines $x=a$ & $x=b$. we know that area of trapezium = height × (avg. of parallel lines).



Area = $\frac{1}{2} \times h \times (a+b.)$

→ So if we join the points $(a, f(a))$ & $(b, f(b))$, it forms a trapezium, with parallel side lengths as $f(a)$ & $f(b)$. height of trapezium is $(b-a)$, Hence from the following graph, area of trapezium is $\frac{b-a}{2}[f(a)+f(b)]$.

$$\frac{\delta x}{2}[f(a)+f(b)]$$



Q.5 Explain Basic Principal (No Derivation) in deriving Newton Cote's integration formulas.

→ The basic principal in obtaining Newton Cote's integration formulas is to fix abscissas $x_0, x_1, x_2, \ldots x_n \in [a, b]$ beforehand & integrand $f(x)$ is approximated by interpolating polynomial $P_n(x)$, $(P_n(x_i) = y_i,\ y_i = 1, 2, \ldots n)$

~~Interpolat~~ Integral of $P_n(x)$ is taken as approximate value of $I(f)$ formula we derive, is of the form $\sum_{i=0}^{n} w_i f(x_i)$

→ As $f(x)$ is being approximated by interpolating polynomial $P_n(x)$ of degree n, $f(x) \simeq P_n(x)$.

$\therefore P_n(x_i) = f(x_i)$ & $I(f) = \int_a^b f(x)\,dx \simeq \int_a^b P_n(x)\,dx$

→ Interpolating polynomial $P_n(x)$ is given by

$P_n(x) = \sum_{i=0}^{n} l_i(x)\, y_i$, $(y_i = f(x_i))$ where,

$$l_i(x) = \prod_{j=0, j \neq i}^{n} \frac{(x - x_j)}{(x_i - x_j)} \quad \forall\ i = 0, 1, \dots, n$$

Thus, $\int_a^b P_n(x)\,dx = \int_a^b \left(\sum_{i=0}^{n} l_i(x)\, y_i \right) dx$

$$= \sum_{i=0}^{n} \left(\int_a^b l_i(x)\,dx \right) y_i$$

$$= \sum_{i=0}^{n} w_i y_i \ ; \text{ giving } w_i = \int_a^b l_i(x)\,dx$$

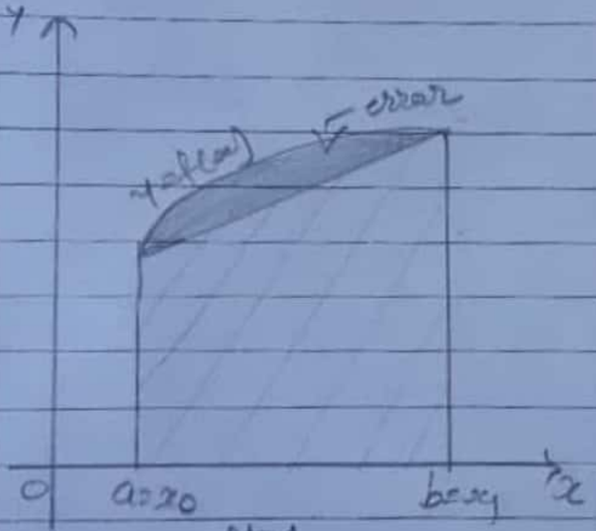
Q6 Write the Composite form of trapezoidal rule for $\int_a^b f(x)\,dx$.

→ If $[a, b]$ is divided into N equal subdivisions. Show graphically, how use of Composite form reduces the truncation error.

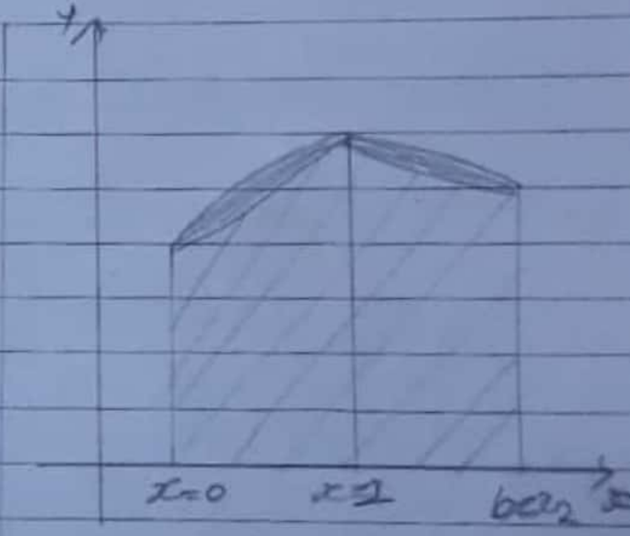
Ans. If [a,b] is divided into N equal subdivisions, the composite form of trapezoidal rule for $\int_a^b f(x)\,dx$ is,

$$\int_a^b f(x)\,dx \simeq \frac{h}{2}\left[f(x_0) + 2\sum_{i=1}^{N-1} f(x_i) + f(x_N)\right]$$

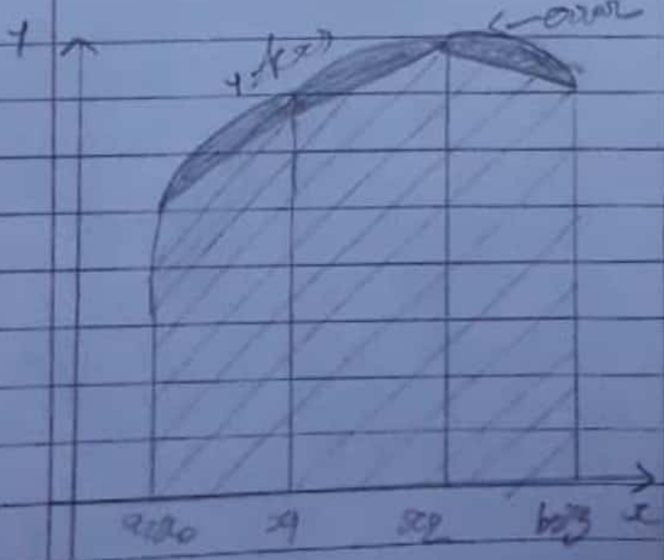
→ The following fig.s shows, as number of segments of intervals are increasing, i.e., we of composite form reduces truncation error.



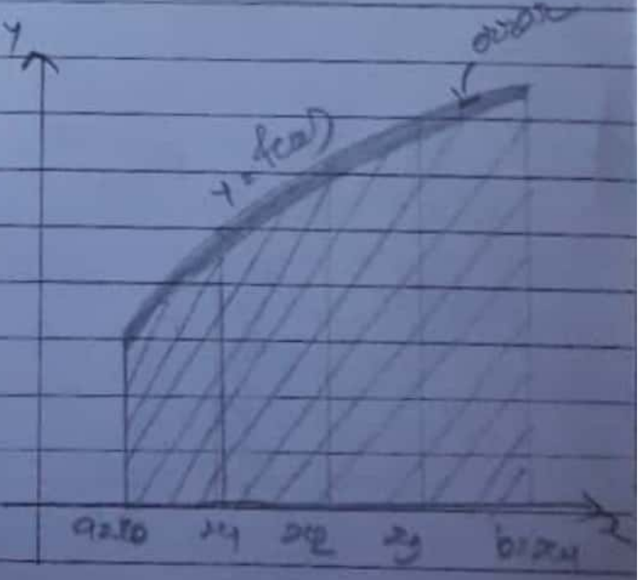
N=1



N=2



N=3



N=4

Q.7 Estimate $\int_0^1 e^{-x^2} dx$ using trapezoidal rule by taking.

i) 2 subdivision:

Ans. Here $N = 2 \Rightarrow h = \frac{b-a}{N} = \frac{1-0}{2} = 0.5$

$x_0 = 0 \Rightarrow f(x_0) = e^0 = 1$
$x_1 = 0.5 \Rightarrow f(x_1) = e^{-0.5^2} = e^{-0.25} = 0.7788$
$x_2 = 1 \Rightarrow f(x_2) = e^{-1^2} = e^{-1} = 0.3679$

$$\int_0^1 e^{-x^2} dx = \int_0^1 f(x)dx = \frac{h}{2}\left[f(x_0) + 2f(x_1) + f(x_2)\right]$$

$$= \frac{0.5}{2}\left[1 + 2(0.7788) + 0.3679\right]$$

$$= 0.25[2.9255]$$

$$\boxed{\int_0^1 e^{-x^2} dx = 0.7313}$$

ii) 4 subdivisions

$\therefore N = 4 \Rightarrow h = 1-0/4 = 0.25$

$x_0 = 0 \Rightarrow f(x_0) = e^0 = 1$
$x_1 = 0.25 \Rightarrow f(x_1) = e^{-0.25^2} = e^{-0.0625} = 0.9394$
$x_2 = 0.50 \Rightarrow f(x_2) = e^{-0.50^2} = 0.7788$
$x_3 = 0.75 \Rightarrow f(x_3) = e^{-0.75^2} = 0.5698$
$x_4 = 1 \Rightarrow f(x_4) = e^{-1} = 0.3679$

$$\int_0^1 f(x)\,dx = h/2\left[f(x_0) + 2\sum_{i=1}^{3} f(x_i) + f(x_4)\right]$$

$$\int_0^1 e^{-x^2}\,dx = \frac{0.25}{2}\left[1 + 2(0.9394 + 0.7788 - 0.5696) + 0.3679\right]$$

c) 6 Subdivisions.

Ans. Here $N = 6$, $h = 1-0/6 = 0.1667$

$x_0 = 0$ $\quad 0/6 \Rightarrow f(x_0) = e^{-0} = 1$

$x_1 = 0.1667$ or $1/6 \Rightarrow f(x_1) = e^{-(1/6)^2} = 0.9726$

$x_2 = 0.3334$ $\quad 2/6 \Rightarrow f(x_2) = e^{-(2/6)^2} = 0.8948$

$x_3 = 0.5001$ $\quad 3/6 \Rightarrow f(x_3) = e^{-(3/6)^2} = 0.7788$

$x_4 = 0.6668$ $\quad 4/6 \Rightarrow f(x_4) = e^{-(4/6)^2} = 0.6412$

$x_5 = 0.8335$ $\quad 5/6 \Rightarrow f(x_5) = e^{-(5/6)^2} = 0.4994$

$x_6 = 1$ $\quad 6/6 \Rightarrow f(x_6) = 0.3679$

$$\int_0^1 f(x)\,dx = h/2\left[f(x_0) + 2\sum_{i=0} f(x_i) + f(x_6)\right]$$

$$\int_0^1 e^{-x^2}\,dx = \frac{0.1667}{2}\left[1 + 2(0.9726 + 0.8948 + 0.7788 + 0.6412 + 0.4994) + 0.3679\right]$$

$$= 0.08334[8.9415]$$

$$\boxed{\int_0^1 e^{-x^2} = 0.7452}$$

Q.8 Calculate approximate integral value of $\int_0^2 f(x)dx$ when $f(x)$ is

a) x^2 using (i) Traperoidal rule

$$\int_a^b f(x)dx = h/_2 [f(a) + f(b)]$$

Here $n=1$ $h = \frac{b-a}{n} = \frac{2-0}{1} = 2.$

$$\therefore \int_0^2 x^2 dx = 2/_2 [f(0) + f(2)] = 1[0+4] = 4$$

(ii) Simpson's 1/3 rule:

$$\rightarrow \int_a^b f(x)dx = h/_3 [f(a) + 4f(a+h) + f(a+2h)]$$

Here, $n=2$ $h = \frac{b-a}{n} = \frac{2-0}{2} = 1$

$$\therefore \int_0^2 x^2 dx = 1/_3 [f(0) + 4(f(0+1)) + f(0+2)]$$
$$= 1/_3 [0 + 4(1) + 4]$$
$$= 1/_3 [8] = 2.6667$$

(iii) Simpson's 3/8 rule:

$$\int_a^b f(x) dx = 3h/_8 [f(a) + 3f(a+h) + 3f(a+2h) + f(a+3h)]$$

Here $n = 3$, $h = 2-0/3 = 2/3$

$$\int_0^2 x^2 dx = \frac{3}{8} \times \frac{2}{3} \left[f(0) + 3f(0+2/3) + 3f(0+2\cdot\frac{2}{3}) + f(0+3\cdot 2/3) \right]$$

$$= \frac{1}{4} \left[0 + 3 \times \frac{4}{9} + 3 \times \frac{16}{9} + 4 \right]$$

$$= \frac{4}{4} \left[\frac{1}{3} + \frac{4}{3} + 1 \right]$$

$$= \boxed{2.6667}$$

b) x^4

i) Trapezoidal rule, $\int_a^b f(x)dx = h/2 \, (f(a) + f(b))$

Here $n = 1$, $h = 2/1 = 2$

$$\int_0^2 x^4 dx = 2/2 \left[f(0) + f(2) \right] = 1 \left[0 + 16 \right] = 16$$

ii) Simpson's 1/3 rule $n = 2$, $h = 1$

$$\int_0^2 x^4 dx = \frac{1}{3} \left[f(0) + 4(f(0+1)) + f(0+2) \right]$$

$$= \frac{1}{3} \left[0 + 4 + 16 \right]$$

$$= \boxed{6.6667}$$

iii) Simpson's 3/8 rule:

$$\int_a^b f(x)dx = 3h/8 \left[f(a) + 3f(a+h) + 3f(a+2h) + f(a+3h) \right]$$

Here $n=3$, $h=2/3$

$$\int_0^2 x^4\,dx = \frac{1}{4}\left[f(0)+3f(\tfrac{2}{3})+3f(2\cdot\tfrac{2}{3})+f(3\cdot\tfrac{2}{3})\right]$$

$$=\frac{1}{4}\left[0+3\times\frac{16}{81}+3\times\frac{256}{81}+16\right]$$

$$=\frac{1}{4}\left[\frac{16}{27}+\frac{256}{27}+16\right]$$

$$=\frac{16}{4}\left[\frac{1}{27}+\frac{16}{27}+1\right]$$

$$=4\left(\frac{44}{27}\right)$$

$$=\boxed{6.5185}$$

(C) $1/(x+1)$

i) Trapezoidal, $n=1$, $h=2$.

$$\int_0^2 \frac{1}{(x+1)}\,dx = \frac{2}{2}\left[f(0)+f(2)\right]$$

$$=1\left[1+0.3333\right]$$

$$=\boxed{1.3333}$$

ii) Simpson's $1/3$ rule, $n=2$, $h=1$

$$\int_0^2 \frac{1}{(x+1)}\,dx = \frac{h}{3}\left[f(0)+4f(0+1)+f(0+2)\right]$$

$$=\frac{1}{3}\left[1+\overset{2}{\cancel{4}}\times\frac{1}{\cancel{2}}+0.3333\right]$$

$$=\frac{1}{3}\left[3.3333\right]$$

$$=\boxed{1.1111}$$

ii) Simpson's $3/8$ rule $n=3$, $h=2/3$

$$\int_0^2 \frac{1}{(x+1)}\,dx = \frac{3h}{8}\left[f(0)+3f(0+2/3)+3f(0+2\cdot 2/3)+f(0+3\cdot 2/3)\right]$$

$$= \frac{1}{4}\left[1+3\times\left(\frac{3}{5}\right)+3\times\left(\frac{3}{7}\right)+\frac{1}{3}\right]$$

$$= \frac{1}{4}\left[1+1.8+1.2858+0.3333\right]$$

$$= \boxed{1.1048}$$

Here $f(4/3) = \dfrac{1}{4/3+1} = \dfrac{1}{\frac{4+3}{3}} = 3/7.$

$f(6/3) = f(2) = \dfrac{1}{2+1} = \dfrac{1}{3} = 0.3333.$

(d) $\sqrt{1+x^2}$

(i) Trapezoidal rule $n=1$, $h=2$

$$\int_0^2 \sqrt{1+x^2}\,dx = h/2\left[f(0)+f(2)\right]$$

Here, $f(0) = 0.$

$f(2) = \sqrt{1+2^2} = \sqrt{5} = 2.2361$

$$\int_0^2 \sqrt{1+x^2}\,dx = 2/2\left[0+2.2361\right]$$

$$= \boxed{2.2361.}$$

iii) Simpson's $3/8$ rule; $n=3$, $h=2/3$

$$\int_0^2 \sqrt{1+x^2} = \frac{3h}{8}\left[f(0)+3f(0+\tfrac{2}{3})+3f(0+2\cdot\tfrac{2}{3})+f(0+3\cdot\tfrac{2}{3})\right]$$

Here. $f(0)=0$, $f(2/3)=\sqrt{1+4/9} = \sqrt{13/9} = 1.2019$.

$f(4/3) = \sqrt{1+12/9} = \sqrt{25/9} = 5/3 = 1.6667$.

$$\int_0^2 \sqrt{1+x^2}\,dx = \frac{1}{4}\left[0+3\times 1.2019+3\times 1.6667+2.2361\right]$$

$$= \frac{1}{4}\left[10.8419\right]$$

$$= \boxed{2.7105}$$

2) $\sin x$

i) Trapezoidal, $n=1$, $h=2$.

$$\int_0^2 \sin x\,dx = \frac{h}{2}\left[f(0)+f(2)\right]$$

Here $f(0) = \sin 0 = 0$, $f(2) = \sin 2 = 0.9093$

$$\int_0^2 \sin x\,dx = 1\left[0+0.9093\right] = 0.9093$$

ii) Simpson's $1/3$ rule, $n=2$, $h=1$

$$\int_0^2 \sin x\,dx = \frac{h}{3}\left[f(0)+4f(0+1)+f(0+2)\right]$$

Here $f(1) = 0.8415$.

$$\int_0^2 \sin x \, dx = \frac{1}{3}\left[0 + 4 \times 0.8415 + 0.9093\right]$$
$$= \frac{1}{3}\left[4.2753\right]$$
$$= 1.4251$$

iii) Simpson's $3/8$ rule, $n = 3$, $h = 2/3$

$$\int_0^2 \sin x \, dx = \frac{3h}{8}\left[f(0) + 3f(0+2/3) + 3f(0+2 \cdot 2/3) + f(0+3 \cdot 2/3)\right]$$

Here, $f(2/3) = \sin 2/3 = 0.6184$.
$f(4/3) = \sin 4/3 = 0.9720$

$$\int_0^2 \sin x \, dx = \frac{1}{4}\left[0 + 3 \times 0.6184 + 3 \times 0.9720 + 0.9093\right]$$
$$= \frac{1}{4}\left[5.6805\right]$$
$$= \boxed{1.4201}$$

f) e^x

i) Trapezoidal rule, $n = 1$, $h = 2$

$$\int_0^2 e^x \, dx = h/2\left[f(0) + f(2)\right]$$

Here, $f(0) = e^0 = 1$ $f(2) = e^2 = 7.3891$.

$$\int_0^2 e^x \, dx = 1\left[1 + 7.3891\right] = \boxed{8.3891}$$

Page: 18
Date: / /

ii) Simpson's 1/3 rule, $n=2$, $h=1$

$$\int_0^2 e^x\,dx = \frac{h}{3}\left[f(0) + 4f(0+1) + f(0+2)\right]$$

Here, $f(1) = e^1 = 2.7183$

$$\int_0^2 e^2\,dx = \frac{1}{3}\left[1 + 4\times 2.7183 + 7.3891\right]$$

$$= \frac{1}{3}\left[19.2623\right]$$

$$= \boxed{6.4208}$$

iii) Simpson's 3/8 rule, $n=3$, $h=2/3$

$$\int_0^2 e^x\,dx = \frac{3h}{8}\left[f(0) + 3f(0+2/3) + 3f(0+2\cdot 2/3) + f(0+3\cdot 2/3)\right]$$

Here, $f(2/3) = e^{2/3} = 1.9477$, $f(4/3) = e^{4/3} = 3.7937$

$$\therefore \int_0^2 e^x\,dx = \frac{1}{4}\left[1 + 3\times 1.9477 + 3\times 3.7937 + 7.3891\right]$$

$$= \frac{1}{4}(25.6133)$$

$$= \boxed{6.4033}$$

Q.9 Given function f at the following values

x	1.8	2.0	2.2	2.4	2.6
$f(x)$	3.12014	4.42569	6.04241	8.03014	10.46075

Approximate $\int_{1.8}^{2.6} f(x)\,dx$ using

(a) Trapezoidal rule (b) Simpson's rule

→ Here $N=4$, $h = \frac{2.6-1.8}{4} = 0.2$

(a) Trapezoidal rule:

$$\int_a^b f(x)dx = h/2\left[f(x_0) + 2\sum_{i=1}^{N-1} f(x_i) + f(x_N)\right]$$

$$\int_{1.8}^{2.6} f(x)\,dx = \frac{0.2}{2}\left[3.12014 + 2(4.42569 + 6.04241 + 8.03014) + 10.48875\right]$$

$$= 0.1\left[50.58387\right]$$

$$= \boxed{5.05534}$$

(b) Simpson's 1/3 rule

$$\int_a^b f(x)\,dx = h/3\left[f(x_0) + 4\left[f(x_1) + f(x_3)\right] + 3\left(f(x_2)\right) + f(x_4)\right]$$

$$\int_{1.8}^{2.6} f(x)dx = \frac{0.2}{3}\left[3.12014 + 4\left[4.42569 + 8.03014\right] + 3(4.42569) + 10.46675\right]$$

$= 0.06667 \, [76.68728]$

$$\int_{1.8}^{2.6} f(x)\,dx = \boxed{5.11274}$$

Q.10 Given the form of closed type Newton Cote's integration formulas as:

$$\int_a^b f(x)\,dx \simeq nh \sum_{i=0}^{n} C_i^n f(x_i) \text{ with } h = \frac{b-a}{n}, \; x_i = x_0 + ih,$$

$$i = 0, 1, \ldots n.$$

State two properties of Newton Cote's Cofficients C_i^n.

→ C_i^n have two important properties

i) $\sum_{i=0}^{n} C_i^n = 1$

ii) $C_i^n = C_{n-i}^n$

Q.11 Derive :

a) Trapezoidal Rule :

→ we have the closed type Newton Cote's integration formulas

$$\int_a^b f(x)dx \simeq nh \sum_{i=0}^{n} C_i^n f(x_i) \text{ with } h = \frac{b-a}{n},$$

$$x_i = x_0 + ih, \; i = 0,1,2,n.)$$

let $n=1$

$$\int_a^b f(x)\,dx = 1h \sum_{i=0}^{1} C_i^n f(x_i)$$

$$= hC_0^1 f(x_0) + hC_1^1 f(x_1)$$

By two properties of Newton Cote's Cofficients

$C_0^1 = C_1^1$ & $C_0^1 + C_1^1 = 1$ gives $C_0^1 = C_1^1 = 1/2$

$\Rightarrow \int_a^b f(x)dx = h/2\,[f(x_0) + f(x_1)]$, which trapezoidal formula.

b) Simpson's 1/3 Rule :

we have,

$$\int_a^b f(x)\,dx \simeq nh \sum_{i=0}^{n} C_i^n f(x_i) \text{ with } h = \frac{b-a}{2},$$

$$x_i = x_0 + ih, \; i = 0,1,2..n.$$

let $n = 2$

$$\int_a^b f(x)\,dx = 2h \sum_{i=0}^{2} C_i^2 f(x_i) \quad \text{--- (1)}$$

Here $C_0^2 = C_2^2$ & $C_0^2 + C_1^2 + C_2^2 = 1$

By evaluating C_2^2

w.k.t. $C_i^n = \dfrac{(-1)^{n-i}}{n\,i!\,(n-i)!} \displaystyle\int_0^n \frac{t(t-1)\ldots(t-n)}{(t-i)}\,dt.$

$$\therefore C_2^2 = \frac{(-1)^{2-2}}{2\,2!\,(2-2)!} \int_0^2 \frac{t(t-1)(t-2)}{(t-2)}\,dt.$$

$$= \frac{1}{4} \int_0^2 t(t-1)\,dt$$

$$= \frac{1}{4} \int_0^2 (t^2 - t)\,dt = \frac{1}{4}\left[\frac{t^3}{3} - \frac{t^2}{2}\right]_0^2$$

$$= \frac{1}{4}\left[\frac{8}{3} - \frac{4}{2}\right] = \frac{1}{4}\left(\frac{8}{3} - 2\right) = \frac{1}{4}\left(\frac{8-6}{3}\right)$$

$$= \frac{2}{4\times 3} = \frac{1}{6}$$

$$\therefore C_2^2 = \frac{1}{6} \Rightarrow C_0^2 = C_2^2 = \frac{1}{6}$$

& $C_1^2 = 1 - (C_0^2 + C_2^2) = 1 - \frac{2}{6} = \frac{4}{6}$

$$\therefore (1) \Rightarrow \int_a^b f(x)\,dx \simeq 2h\,C_0^2 f(x_0) + 2h\,C_1^2 f(x_1) + 2h\,C_2^2 f(x_2)$$

$$\simeq \frac{h}{3} f(x_0) + \frac{4h}{3} f(x_1) + \frac{h}{3} f(x_2)$$

$$\Rightarrow \int_a^b f(x)\,dx \simeq \frac{h}{3}\left[f(x_0) + 4f(x_1) + f(x_2)\right]$$

which can also be expressed as

$$\int_a^b f(x)\,dx = \frac{b-a}{6}\left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right]$$

$$\int_a^b f(x)\,dx \simeq h/3\,[f(a) + 4f(a+h) + f(a+2h)]$$

This is Simpson's 1/3 Rule.

3) Simpson's 3/8 Rule.

we have,

$$\int_a^b f(x)\,dx \simeq nh \sum_{i=0}^{n} C_i^n f(x_i) \text{ with } h = \frac{b-a}{2},$$

$x_i = x_0 + ih$; $i = 0, 1, n$.

let $n = 3$

$$\int_a^b f(x)\,dx = 3h \sum_{i=0}^{3} C_i^3 f(x_i) \quad \text{---(1)}$$

we know that $C_0^3 = C_3^3$ & $C_1^3 = C_2^3$ & $\sum_{i=0}^{3} C_i^3 = 1$

$$\therefore 2C_0^3 + 2C_1^3 = 1$$

$$\text{Cok. } C_i^n = \frac{(-1)^{n-i}}{ni!(n-i)!} \int_0^n \frac{t(t-1)\ldots(t-n)}{(t-i)}\,dt$$

$$\therefore C_3^3 = \frac{(-1)^{3-3}}{3\cdot 3!(3-3)!} \int_0^3 \frac{t(t-1)(t-2)\cancel{(t-3)}}{\cancel{(t-3)}}$$

$$= \frac{1}{18}\int_0^3 t(t-1)(t-2)\,dt$$

$$= \frac{1}{18}\int_0^3 t(t^2-3t+2)\,dt$$

$$= \frac{1}{18}\int_0^3 (t^3-3t^2+2t)\,dt$$

$$= \frac{1}{18}\left[\frac{t^4}{4}-\frac{3t^3}{3}+\frac{2t^2}{2}\right]_0^3$$

$$= \frac{1}{18}\left[\frac{81}{4}-27+9\right] = \frac{1}{18}\left[\frac{9}{4}\right] = \frac{1}{8}$$

$$\therefore C_3^3 = \frac{1}{8} = C_0^3 \ \& \ C_1^3 = \frac{1-2C_0^3}{2} = \frac{3}{4} = C_2^3$$

$$\therefore (1) \Rightarrow \int_a^b f(x)\,dx = \frac{3h}{8}\left[f(x_0)+3f(x_1)+3f(x_2)+f(x_3)\right]$$

$$\int_a^b f(x)\,dx = \frac{3h}{8}\left[f(a)+3f(a+h)+3f(a+2h)+f(a+3h)\right]$$

Q.12 Estimate $\int_0^4 e^x\,dx$ by

a) Simpson's 1/3 rule.

By Simpson's 1/3 rule.

$$\int_a^b f(x)\,dx = \frac{h}{3}\left[f(a)+4f(a+h)+f(a+2h)\right]$$

Here $n=2$, $h = \frac{4-0}{2} = 2$.

$$\therefore \int_0^4 e^x dx = \frac{2}{3}\left[f(0) + 4f(0+2) + f(0+2\cdot2)\right]$$

$$= \frac{2}{3}\left[1 + 4(7.38906) + 54.59815\right]$$

$$= \frac{2}{3}\left[85.15439\right]$$

$$= \boxed{56.769593}$$

(b) Composite Simpson's 1/3 rule with N = 2

Ans. here $N = 2$, $h = \frac{4-0}{2} = 2$.

$x_0 = 0 \Rightarrow f(x_0) = e^0 = 1$
$x_1 = 2 \Rightarrow f(x_1) = e^2 = 7.38906$
$x_2 = 4 \Rightarrow f(x_2) = e^4 = 54.59815$

$$\therefore \int_0^4 e^x dx = \frac{h}{3}\left[f(x_0) + 2(f(x_1)) + f(x_2)\right]$$

$$= \frac{2}{3}\left[1 + 4(7.38906) + 54.59815\right]$$

$$= \frac{2}{3}\left[85.15439\right]$$

$$= \boxed{56.769593}$$

(c) Composite Simpson's 1/3 rule with N = 4

Ans. Here $N = 4$ $h = \frac{4-0}{4} = 1$

$\therefore x_0 = 0 \Rightarrow f(x_0) = 1$

$x_1 = 1 \Rightarrow f(x_1) = 2.71828$

$x_2 = 2 \Rightarrow f(x_2) = 7.38906$

$x_3 = 3 \Rightarrow f(x_3) = 20.08534$

$x_4 = 4 \Rightarrow f(x_4) = 54.59815$

$$\int_0^4 e^x dx = \frac{h}{3}\left[f(x_0) + 4(f(x_1) + f(x_3)) + 2(f(x_2)) + f(x_4)\right]$$

$$= \frac{1}{3}\left[1 + 4(2.71828 + 20.08554) + 2(7.38906) + 54.59815\right]$$

$$= \frac{1}{3}\left[161.59155\right]$$

$$\int_0^4 e^x dx = \boxed{53.86385}$$

(d) Calculate error in above (exact value $= e^4 - 1$) $= 53.59815$

error in (a) Simpson's $\frac{1}{3}$ rule

error = True value − Approximate value

$= 53.59815 - 56.76993$

$= \underline{-3.171449}$

error in (b) Composite Simpson's rule with $N = 2$

$\therefore$ error $= 53.59815 - 56.769593$

$= \underline{-3.171445}$

error in ⓒ Composite Simpson's rule with $N=4$

$$\text{error} = 53.59815 - 53.86385$$
$$= -0.2657$$

Q.14 A Car laps a race track in 84 Seconds. The Speed of the car at each 6-Second interval is determined by using a radar gun & is given from the begining of the lap, in feet/second by the entries in the following table.

Time	0	6	12	18	24	30	36	42	48	54
speed	124	134	148	156	147	133	121	109	99	85

60	66	72	78	84
78	89	104	116	123

ii) How long is the track.

→ Here to find How long is the track, we have to find

$$\int_0^{84} f(x)\,dx$$

By Composite Simpsons $\frac{1}{3}$ formulla

Here $N=14$ $h = 84/14 = 6$

$$\therefore \int_0^{84} f(x)\,dx = \frac{h}{3}\Big[f(x_0) + 4(f(x_1) + f(x_3) + f(x_5) + f(x_7) + f(x_9) + f(x_{11}) + f(x_{13})) + 2(f(x_2) + f(x_4) + f(x_6) + f(x_8) + f(x_{10}) + f(x_{12})) + f(x_{14})\Big]$$

$$= \frac{6}{3}\Big[124 + 4(134 + 156 + 133 + 109 + 85 + 89 + 116) + 2(148 + 147 + 121 + 99 + 78 + 109) + 123\Big]$$

$$= 2[4929]$$

$$= \boxed{9858 \text{ feet}}$$

Q.15 Approximate $\int_0^2 x^2 \ln(x^2+1)\,dx$ using $h = 0.25$ use

1) Composite trapezoidal rule.

Here $h = 0.25$, $N = \frac{b-a}{0.25} = \frac{2}{0.25} = 8$.

$$\int_0^2 x^2 \ln(x^2+1)\,dx = \frac{h}{2}\Big[f(x_0) + 2(f(x_1) + f(x_2) + f(x_3) + f(x_4) + f(x_5) + f(x_6) + f(x_7)) + f(x_8)\Big]$$

Here,

$x_0 = 0 \Rightarrow f(x_0) = 0$

$x_1 = 0.25 \Rightarrow f(x_1) = 0.00379$

$x_2 = 0.50 \Rightarrow f(x_2) = 0.05579$

$x_3 = 0.75 \Rightarrow f(x_3) = 0.25104$

$x_4 = 1 \Rightarrow f(x_4) = 0.69315$

$x_5 = 1.25 \Rightarrow f(x_5) = 1.47019$

$x_6 = 1.50 \Rightarrow f(x_6) = 2.65197$
$x_7 = 1.75 \Rightarrow f(x_7) = 4.29301$
$x_8 = 2 \Rightarrow f(x_8) = 6.43775$

$$\int_0^2 x^2 \ln(x^2+1)\,dx = \frac{0.25}{2}\left[0 + 2(0.00379 + 0.05579 + 0.25104 + 0.69315 + 1.47029 + 2.65197 + 4.29301) + 6.43775\right]$$

$$= 0.125\,[25.27583]$$

$$\int_0^1 x^2 \ln(x^2+1)\,dx = \underline{3.1595}$$

(b) Composite Simpson's $\frac{1}{3}$ Rule :

Here $h = 0.25$, $N = 8$.

$$\int_0^2 x^2 \ln(x^2+1)\,dx = \frac{h}{3}\left[f(x_0) + 4\{f(x_1) + f(x_3) + f(x_5) + f(x_7)\} + 2\{f(x_2) + f(x_4) + f(x_6)\} + f(x_8)\right]$$

$$= \frac{0.25}{3}\left[0 + 4(0.00379 + 0.25104 + 1.47029 + 4.29301) + 2(0.05379 + 0.69315 + 2.65197) + 6.43775\right]$$

$$= 0.083334\,[37.31207]$$

$$\int_0^2 x^2 \ln(x^2+1)\,dx = \underline{3.10921}$$

(c) Composite midpoint rule :

Here , midpoint rule is , $\int_a^b f(x)\,dx \cancel{+ (b-a)}$

$$\int_a^b f(x)\,dx = (b-a)\, f\left(\frac{a+b}{2}\right)$$

$$\therefore \int_0^2 x^2 \ln(x^2+1)\,dx = \int_0^{0.25} x^2 \ln(x^2+1)\,dx + \int_{0.25}^{0.10} x^2 \ln(x^2+1)\,dx +$$

$$\cdots + \int_{1.75}^{2} x^2 \ln(x^2+1)\,dx$$

$$\int_0^{0.25} x^2 \ln(x^2+1)\,dx = (0.25-0)\; f\left(\frac{0.25+0}{2}\right)$$

$$= 0.25\,(f(0.121))$$

$$= 0.25 \times 0.00024$$

$$= 0.00006.$$

$$\int_{0.25}^{0.50} x^2 \ln(x^2+1)\,dx = 0.25 \times f(0.375)$$

$$= 0.25 \times 0.01850$$

$$= 0.00463.$$

$$\int_{0.50}^{0.71} x^2 \ln(x^2+1)\,dx = 0.25 \times f(0.625)$$

$$= 0.25 \times 0.12881$$

$$= 0.03220.$$

$$\int_{0.75}^{1} x^2 \ln(x^2+1)\,dx = 0.25 \times f(0.875)$$

$$= 0.25 \times 0.43526$$

$$= 0.10882.$$

$$\int_{1}^{1.25} x^2 \ln(x^2+1)\,dx = 0.25 \times f(1.25)$$
$$= 0.25 \times 1.03509$$
$$= 0.25877$$

$$\int_{1.25}^{1.50} x^2 \ln(x^2+1)\,dx = 0.25 \times f(1.375)$$
$$= 0.25 \times 2.00685$$
$$= 0.50171.$$

$$\int_{1.50}^{1.75} x^2 \ln(x^2+1)\,dx = 0.25 \times f(1.625)$$
$$= 0.25 \times 3.41210$$
$$= 0.85302$$

$$\int_{1.75}^{2} x^2 \ln(x^2+1)\,dx = 0.25 \times f(1.875)$$
$$= 0.25 \times 5.29996$$
$$= 1.32499.$$

$$\int_{0}^{2} x^2 \ln(x^2+1)\,dx = 0.00006 + 0.00463 + 0.03220 + 0.10882 + 0.25877 + 0.50171 + 0.85302 + 1.32499.$$

$$\int_{0}^{2} x^2 \ln(x^2+1)\,dx = 3.0842.$$

Q.16 Approximate $\int_0^2 x^2 e^{-x^2} dx$ using $h = 0.25$ we

a) Composite Traperoidal rule.

Here $h = 0.25$, $N = 8$.

$x_0 = 0$ $f(x_0) = 0$
$x_1 = 0.25$ $f(x_1) = 0.05871$
$x_2 = 0.50$ $f(x_2) = 0.1947$
$x_3 = 0.75$ $f(x_3) = 0.3205$
$x_4 = 1$ $f(x_4) = 0.36788$
$x_5 = 1.25$ $f(x_5) = 0.3275$
$x_6 = 1.50$ $f(x_6) = 0.23715$
$x_7 = 1.75$ $f(x_7) = 0.14324$
$x_8 = 2$ $f(x_8) = 0.07326$

$$\int_0^2 x^2 e^{-x^2} dx = \frac{0.25}{2}\left[0 + 2(0.05871 + 0.1947 + 0.3205 + 0.36788 + 0.3275 + 0.23715 + 0.14324) + 0.07326\right]$$

$$\Rightarrow \int_0^2 x^2 e^{-x^2} dx = \underline{0.42158}$$

b) Composite Simpson's 1/3 Rule:

$h = 0.25$, $N = 8$.

$$\int_0^2 x^2 e^{-x^2} dx = h/3 \left[f(x_0) + 4(f(x_1) + f(x_3) + f(x_5) + f(x_7)) + 2(f(x_2) + f(x_4) + f(x_6)) + f(x_8)\right]$$

$\Rightarrow$

$$= \frac{0.25}{3}\Big[0+4(0.05871+0.3205+0.3275+0.14324)$$
$$+2(0.1947+0.36788+0.23715)+0.0732\Big]$$

$$= 0.083\,[5.07252]$$

$$\int_0^2 x^2 e^{-x^2}\,dx = \underline{0.42102}$$

(3) Composite midpoint rule

$$\int_3^2 x^2 e^{-x^2}dx = \int_0^{0.25} x^2 e^{-x^2}dx + \int_{0.25}^{0.50} x^2 e^{-x^2}dx + \cdots + \int_{1.75}^{2} x^3 e^{-x^2}dx$$

$$\int_0^{0.25} x^2 e^{-x^2}dx = 0.25 + (0.125)$$
$$= 0.25 \times 0.01538$$
$$= 0.00385$$

$$\int_{0.25}^{0.50} x^2 e^{-x^2}dx = 0.25 \times f(0.375)$$
$$= 0.25 \times 0.12218$$
$$= 0.03054.$$

$$\int_{0.50}^{0.75} x^2 e^{-x^2}dx = 0.25 + f(0.625)$$
$$= 0.25 \times 0.26431$$
$$= 0.06608$$

$$\int_{0.75}^{1} x^2 e^{-x^2}dx = 0.25 \times f(0.875)$$
$$= 0.25 \times 0.35605$$
$$= 0.08901$$

$$\int_{1}^{1.25} x^2 e^{-x^2} dx = 0.25 \times f(1.125)$$
$$= 0.25 \times 0.35699$$
$$= 0.08925$$

$$\int_{1.25}^{1.50} x^2 e^{-x^2} dx = 0.25 \times f(1.375)$$
$$= 0.25 \times 0.28544$$
$$= 0.07136 ,$$

$$\int_{1.50}^{1.75} x^2 e^{-x^2} dx = 0.25 \times f(1.625)$$
$$= 0.25 \times 0.18832$$
$$= 0.04708$$

$$\int_{1.75}^{2} x^2 e^{-x^2} dx = 0.25 \times f(1.875)$$
$$= 0.25 \times 0.10452$$
$$= 0.02619 .$$

$$\int_{0}^{2} x^2 e^{-x^2} dx = 0.00385 + 0.03054 + 0.6608 + 0.05001$$
$$+ 0.08925 + 0.07136 + 0.04708 + 0.02613$$

$$= \underline{0.4233}$$

Assignment . 2

Q1 Write Gauss legedre Quarature formula for estimating $\int_{-1}^{1} f(x)dx$ for $n=1, 2$ & 3. (n = number of abscissas).

→ Gauss legedre quarture formula for estimating $\int_{-1}^{1} f(x)dx$

for $n=1$ is

$$\int_{-1}^{1} f(x)dx \cong 2f(0)$$

for $n=2$

$$\int_{-1}^{1} f(x)dx \cong f(-1/\sqrt{3}) + f(1/\sqrt{3})$$

for $n=3$

$$\int_{-1}^{1} f(x)dx \cong \frac{5}{9} f(\sqrt{-3/5}) + 8/9\, f(0) + \frac{5}{9} f(\sqrt{3}/5)$$

Q.2 Approximate $\int_{-1}^{1} e^{x} \cos x\, dx$ using Gauss legendre Quarature formula

i) taking $n=2$

→ The Gauss legedre quarature formula for $\int_{-1}^{1} f(x)dx$ for $n=2$ is

$$\int_{-1}^{1} f(x)\,dx \simeq f\left(-\tfrac{1}{\sqrt{3}}\right) + f\left(\tfrac{1}{\sqrt{3}}\right)$$

$$\therefore \int_{-1}^{1} e^{x}\cos x\,dx \simeq e^{-1/\sqrt{3}}\cos\left(\tfrac{-1}{\sqrt{3}}\right) + e^{1/\sqrt{3}}\cos\left(\tfrac{1}{\sqrt{3}}\right)$$

$$= 1.96297.$$

ii) for $n=3$.

$$\int_{-1}^{1} f(x)\,dx \simeq \tfrac{5}{9} f\left(-\sqrt{\tfrac{3}{5}}\right) + \tfrac{8}{9} f(0) + \tfrac{5}{9} f\left(\sqrt{3/5}\right)$$

$$\int_{-1}^{1} e^{x}\cos x = \tfrac{5}{9} \times e^{-\sqrt{3/5}} \cos\left(\sqrt{3/5}\right) + \tfrac{8}{9} e^{0}\cos 0 + \tfrac{5}{9} e^{\sqrt{3/5}} \times \cos\left(\sqrt{3/5}\right)$$

$$= 1.93339$$

Q.3 Show, how an integral $\int_a^b f(x)\,dx$ over an arbitrary $[a,b]$ can be transformed into integral over $[-1,1]$.

→ To transform $\int_a^b f(x)\,dx$ to $\int_{-1}^{1} g(t)\,dt$ we use change of variables from x to t defined by $x = \frac{b-a}{2}t + \frac{b+a}{2}$.

→ on putting $x=a$ & solving for t we obtain, $a = \frac{b-a}{2}t + \frac{b+a}{2}$.

which on solving gives value of t as -1. Similarly, putting $x=b$, we obtain, $b = \frac{b-a}{2}t + \frac{b+a}{2}$ which on solving

gives $t = 1$. Thus limits of integration change from a to b into -1 to 1.
now also $dx = \frac{b-a}{2} dt$.

$$\therefore \int_a^b f(x)dx = \frac{b-a}{2} \int_{-1}^{1} f\left(\frac{b-a}{2}t + \frac{b+a}{2}\right) dt.$$

Q.4 Write open Newton Cotes integral formulas for

a) $n = 0$ (midpoint rule)
b) $n = 1$

And show midpoint rule is same as gauss quadrature one point formula

→ open Newton Cotes integral formulas for

a) $n=0$.
midpoint rule, $n=0$, $h = (b-a)/2$

let $a = x_{-1}$, $b = x_1$

$$\int_{x_{-1}}^{x_1} f(x)dx = 2h\, f(x_0) + \frac{1}{3} h^3 f^2(c)$$

Here last term is error term.

b) $n = 1$, $h = (b-a)/3$

$a = x_{-1}$, $b = x_2$

$$\int_{x_{-1}}^{x_2} f(x)dx = \frac{3h}{2}\left[f(x_0) + f(x_1)\right] + \frac{3h^3}{4} f^2(\lambda)$$

Here last term is error term

→ The gauss quadrature one point formula is:

$$\int_a^b f(x) = \omega_1 f(x_1)$$

∴ It gives exact result, when $f(x)$ is of degree $\leq 2n - 1 = 1$

now let $f(x) = 1$ & $f(x) = x$ one by one

for $f(x) = 1$, $\int_a^b 1 dx = \omega_1$

∴ $\omega_1 = b - a$ — (1)

for $f(x) = x$, $\int_a^b x dx = \omega_1 x_1$

$\Rightarrow \left[\frac{x^2}{2}\right]_a^b = \omega_1 x_1$

$\Rightarrow \omega_1 x_1 = \frac{1}{2}(b^2 - a^2)$ — (2)

~~putting resulting gaussian quadrature formula is~~

Putting ① in ②, $x_1 = \frac{1}{2}(b+a)$

∴ The resulting gaussing quadrature formula is

$$\int_a^b f(x)dx \cong (b-a)\, f\left(\frac{a+b}{2}\right)$$

Which is same as midpoint rule.

Q.5 Consider $\int_1^3 (x^6 - x^2 \sin 2x)\,dx$. Exact value is 317.3442466. Estimate value of integral using.

(1) Simple trapezoidal rule:

Q.5 Consider $\int_{1}^{3} (x^6 - x^2 \sin 2x)\,dx$. Exact value is 317.4266. Estimate value of integral using.

1) Simple trapezoidal rule:

Let $f(x) = (x^6 - x^2 \sin 2x)\,dx$, $n=1$, $h = \frac{3-1}{1} = 2$

$\therefore \int_1^3 f(x)dx = h/2\,[f(3) + f(1)]$

$= 2/2\,[731.5147 + 0.0907]$

$\int_1^3 f(x)dx = 731.6054$

2) Open Newton-Cotes formula for $n=0$, $n=1$

for $n=0$, $h = 3-1/2 = 2/2 = 1$ $f(x) = (x^6 - x^2\sin 2x)$

$\therefore \int_1^3 f(x)dx = 2h\left(f\left(\frac{3+1}{2}\right)\right) = 2f(2) = 2 \times 67.0272$

$= \boxed{134.0544}$

for $n=1$, $h = \frac{b-a}{3} = 2/3$

$\int_1^3 f(x)dx = \frac{3\times h}{2}\left[f\left(\frac{1+2}{3}\right) + f\left(1+2\cdot\frac{2}{3}\right)\right]$

$= 1\,[f(5/3) + f(7/3)]$

$= 1\,[21.9628 + 166.8228]$

$= \underline{188.7856}$

(c) Simpson's 1/3 rule:

$\int_1^3 f(x)dx = h/3 \,(f(a)+4f(a+h)+f(a+2h)]$

Here $f(x) = (x^6 - x^2 \sin \pi x)$, $n=2$, $h = \frac{3-1}{n} = 2/2 = 1$

$\int_1^3 f(x)dx = 1/3 \,[f(1) + 4f(2) + f(3)]$

$= 1/3 \,[0.0907 + 4(67.0272) + 731.5147]$

$= 1/3 \,(997.7142]$

$\int_1^3 f(x)dx = 333.2381$

(d) Gauss - Quadrature two point formula:

w.k.t the formula is

$\int_{-1}^{1} f(x)dx = f(-1/\sqrt{3}) + f(1/\sqrt{3})$ — (a)

Our integral is $\int_1^3 (x^6 - x^2 \sin \pi x)dx$

→ We shall need to transform the given interval to $\int_{-1}^{1} g(t)\,dt$ through below equation

$\therefore \int_a^b f(x)dx = \left(\frac{b-a}{2}\right) \int_{-1}^{1} f\left(\frac{b-a}{2}t + \frac{b+a}{2}\right)dt$

→ Here $a=1$, $b=3$, $f(x) = x^6 - x^2 \sin \pi x$

Subsitude $x = \frac{3-1}{2}t + \frac{3+1}{2}$

$\Rightarrow x = t + 2$

$\therefore \int_1^3 f(x)dx = \frac{(3-1)}{2} \int_{-1}^{1} f(t+2)dt$ — (1)

$= 1 \times \int_{-1}^{1} (t+2)^6 - (t+2)^2 \sin 2(t+2)\, dt.$

eq (a) $\Rightarrow \int_1^3 f(x)dx = 1 \times \left[(-1/\sqrt{3}+2)^6 - (-1/\sqrt{3}+2)^2 \sin 2(2 + 1/\sqrt{3}) + (1/\sqrt{3}+2)^6 - (1/\sqrt{3}+2)^2 \sin 2(2 + 1/\sqrt{3}) \right]$

$= 8.2906 - 0.5709 + 293.1168 + 6.0035.$

$\int_1^3 f(x)dx = 306.82.$

(e) Gauss-Quadrature three point formula

→ 3 point formula is

$\int_{-1}^{1} f(x)dx \simeq 5/9 f(-\sqrt{3}/5) + 8/9 f(0) + 5/9 f(\sqrt{3}/5)$ — (b)

By transforming the given interval to $\int_{-1}^{1} g(t)dt$ we have

eq 1) $\rightarrow \int_1^3 f(x)dx = \frac{(3-1)}{2} \int_{-1}^{1} f(t+2)dt.$

$= 1 \times \int_{-1}^{1} (t+2)^6 - (t+2)^2 \sin 2(t+2)\, dt$

Page : 54
Date: / /

Q.2(b) ⇒

$$\int_1^3 f(x)\,dx = \left[\frac{5}{9}\left[(-\sqrt{3/5}+2)^6 - (-\sqrt{3/5}+2)^2 \sin 2(-\sqrt{3/5}+2)\right]\right.$$

$$\left. + 8/9\left[(0+2)^6 - (0+2)^2 \sin 2(0+2)\right]\right]$$

$$= 5/9\left[(\sqrt{3/5}+2)^6 - (\sqrt{3/5}+2)^2 \sin 2(\sqrt{3/5}+2)\right]$$

$$= 5/9(2.4292) + 8/9(67.0272) + 5/9(461.4028)$$

$$\int_1^3 f(x)\,dx = \underline{\underline{317.2642}}$$

Q.6 List the Legendra polynomials $P_n(x)$ for $n=0,1,2,3$ & find their roots. verify that each $P_n(x)$ is

a) Roots are distinct.
b) Roots lie within interval $(-1,1)$.
c) Roots are symmetric with respect to origin.

→ Legendra polynomials for $n=0,1,2,3$.

for $n=0$, $P_0(x)=1$.
There is no root of $P_0(x)$.

→ for $n=1$, $P_1(x)=x$

Here $\int_{-1}^{1} f(x)dx = \omega_1 f(x_1)$

it should give exact result when $x f(x)$ is of degree 1.

$\therefore$ let $f(x)=1$ $\therefore \int_{-1}^{1} 1\,dx = \omega_1 \Rightarrow \omega_1 = [x]_{-1}^{1} = 2$ —①

let $f(x)=x$ $\therefore \int_{-1}^{1} x\,dx = \omega_1 x_1$

$\Rightarrow \left[\frac{x^2}{2}\right]_{-1}^{1} = \omega_1 x_1$

$\therefore \omega_1 x_1 = 1/2 - 1/2 = 0$

Eq ① $\Rightarrow 2x_1 = 0 \Rightarrow x_1 = 0$.
root of $P_1(x)$ is $x_1=0$ which is distinct, $-1<0<1$ & on origin

$\Rightarrow$ for $n=2$, $P_2(x) = \left(\frac{3x^2-1}{2}\right)$

Here $\int_{-1}^{1} f(x)\,dx = \omega_1 f(x_1) + \omega_2 f(x_2)$

let $f(x)=1 \Rightarrow \omega_1+\omega_2 = \int_{-1}^{1} 1\,dx = [x]_{-1}^{1} = 2$

$f(x) = x \Rightarrow \omega_1 x_1 + \omega_2 x_2 = \int_{-1}^{1} x\,dx = \left[\frac{x^2}{2}\right]_{-1}^{1} = \left[\frac{1}{2}-\frac{1}{2}\right] = 0$

$f(x) = x^2 \Rightarrow \omega_1 x_1^2 + \omega_2 x_2^2 = \int_{-1}^{1} x^2\,dx = \left[\frac{x^3}{3}\right]_{-1}^{1} = 2/3$

$f(x) = x^3 \Rightarrow \omega_1 x_1^3 + \omega_3 x_3^3 = \int_{-1}^{1} x^3\,dx = \left[\frac{x^4}{4}\right]_{-1}^{1} = \frac{1}{4}-\frac{1}{4} = 0$

By solving above equations, we get the roots.

$x_1 = -1/\sqrt{3}$ & $x_2 = 1/\sqrt{3}$

for $n=3$, $P_3(x) = \frac{(5x^3-3x)}{2}$

Here $\int_{-1}^{1} f(x)\,dx = \omega_1 f(x_1) + \omega_2 f(x_2) + \omega_3 f(x_3)$

let

$f(x)=1 \Rightarrow \omega_1+\omega_2+\omega_3 = 2$
$f(x)=x \Rightarrow \omega_1 x_1 + \omega_2 x_2 + \omega_3 x_3 = 0$
$f(x)=x^2 \Rightarrow \omega_1 x_1^2 + \omega_2 x_2^2 + \omega_3 x_3^2 = 2/3$
$f(x)=x^3 \Rightarrow \omega_1 x_1^3 + \omega_2 x_2^3 + \omega_3 x_3^3 = 0$
$f(x)=x^4 \Rightarrow \omega_1 x_1^4 + \omega_2 x_2^4 + \omega_3 x_3^4 = 2/5$
$f(x)=x^5 \Rightarrow \omega_1 x_1^5 + \omega_2 x_2^5 + \omega_3 x_3^5 = 0$

$\therefore$ Solving above equations, roots, x_1, x_2, x_3 are.
$x_1 = -\sqrt{3/5}$, $x_2 = 0$, $x_3 = \sqrt{3/5}$.

→ Here all are distinct, $\forall i = 1, 2, 3$, $-1 < x_i < 1$ & roots are symmetric with respect to origin

→ for $n = 2$, roots, $x_1 = -1/\sqrt{3}$ & $x_2 = 1/\sqrt{3}$

are also distinct, both are b/w -1 to 1 & both are symmetric

Q7 Calculate $\int_3^{3.5} \frac{x}{\sqrt{x^2-4}}\,dx$ using

1) 2 point gauss quadrature formula

→ we need to transform the given integral to $\int_{-1}^{1} g(t)\,dt$.

The two point Gaussian Quadrature formula is

$$\int_{-1}^{1} f(x)\,dx = f(-1/\sqrt{3}) + f(1/\sqrt{3}) \quad \text{--- (1)}$$

w.k.t. $$\int_a^b f(x)\,dx = \frac{(b-a)}{2}\int_{-1}^{1} f\left(\frac{b-a}{2}t + \frac{b+a}{2}\right)dt$$

here $a = 3$, $b = 3.5$, $f(x) = \frac{x}{\sqrt{x^2-4}}$

$$\therefore x = \frac{3.5-3}{2}t + \frac{3.5+3}{2}$$

$$= \frac{0.5}{2}t + \frac{6.5}{2}$$

$$\int_3^{3.5} f(x)\,dx = \frac{(3.5-3)}{2} \int_{-1}^{1} f\left(\frac{0.5}{2}t + \frac{6.5}{2}\right) dt \quad \text{—} ①$$

$$= 0.25 \int_{-1}^{1} f(0.25t + 3.25)\, dt.$$

$$= 0.25 \int_{-1}^{1} \frac{(0.25t + 3.25)}{\sqrt{(0.25t + 3.25)^2 - 4}}\, dt$$

$$① \Rightarrow \; = 0.25 \times \left[\frac{(0.25 \times (-1/\sqrt{3}) + 3.25)}{\sqrt{(0.25 \times -1/\sqrt{3}}^2 - 4 + 3.21)} + \frac{(0.25 \times 1/\sqrt{3} + 3.25)}{\sqrt{(0.25 \times 1/\sqrt{3} + 3.25)^2 - 4)}} \right]$$

$$= 0.25\,[1.3071 + 1.2377]$$

$$= 0.63629.$$

2) 3 Point Gauss Quadrature formula.

3 Point formula is

$$\int_{-1}^{1} f(x)\,dx \simeq 5/9\, f(-\sqrt{3/5}) + 8/9\, f(0) + 5/9\, f(\sqrt{3/5}) \quad \text{—} ②$$

By transforming given interval to $\int_{-1}^{1} g(t)\,dt$ we have,

$$\text{by } ① \Rightarrow \int_3^{3.5} f(x)\,dx = \frac{(3.5-3)}{2} \int_{-1}^{1} (0.25t + 3.25)\, dt.$$

$$= 0.25 \int_{-1}^{1} (0.25t + 3.25)\, dt$$

$$= 0.25 \int_{-1}^{1} \frac{(0.25t\ 3.25)}{\sqrt{(0.25t + 3.21)^2 - 4}}\, dt$$

eq. ② $\Rightarrow$ $$\int_{3}^{3.5} f(x)\,dx = 0.25\left[5/9 \times \frac{(0.25 \times -\sqrt{3/5} + 3.25)}{\sqrt{(0.25 \times (-\sqrt{3/5}) + 3.25)^2 - 4}} + \right.$$

$$8/9 \times \frac{(0.25 \times 0 + 3.25)}{\sqrt{(0.25 \times 0 + 3.25)^2 - 4}} +$$

$$\left. 5/9 \times \frac{(0.25 \times \sqrt{3/5} + 3.25)}{\sqrt{(0.25 \times \sqrt{3/5} + 3.25)^2 - 4}} \right]$$

$$= 0.25\,[0.7347 + 1.1977 + 0.6825]$$

$$= \boxed{0.6362}$$

Q.5 Determine Constant a, b, c, d that will produce a quadrature formula

$$\int_{-1}^{1} f(x)\,dx = af(-1) + bf(1) + f'(-1) + df'(1)$$

That has degree of precision 3.
(method of undetermined cofficient)

→ Using method of undetermined cofficient

let $f(x) = 1$

$\therefore f(-1) = 1, \; f(1) = 1, \; f'(-1) = \frac{d(1)}{dx} = 0, \; f'(1) = \frac{d(1)}{dx} = 0$

$\int_{-1}^{1} dx = [x]_{-1}^{1} = 2 = a \cdot 1 + b \cdot 1 + c \cdot 0 + d \cdot 0$ — (1)

→ let $f(x) = x$

$\therefore f(-1) = -1, \; f(1) = 1, \; f'(-1) = \frac{d(x)}{dx} = 1, \; f'(1) = \frac{d(x)}{dx} = 1$

$\int_{-1}^{1} x\,dx = \left[\frac{x^2}{2}\right]_{-1}^{1} = 0 = a \cdot -1 + b \cdot 1 + c \cdot 1 + d \cdot 1$ — (2)

→ let $f(x) = x^2$

$\therefore f(-1) = 1, \; f(1) = 1, \; f'(-1) = \frac{d(x^2)}{dx} = -2, \; f'(1) = \frac{d(x^2)}{dx} = 2$

$\therefore \int_{-1}^{1} x^2 dx = \left[\frac{x^3}{3}\right]_{-1}^{1} = \frac{2}{3} = a \cdot 1 + b \cdot 1 + c \cdot -2 + d \cdot 2$ — (3)

→ let $f(x) = x^3$

$\therefore f(-1) = -1, \; f(1) = 1, \; f'(-1) = \frac{d(x^3)}{dx} = 3x^2 = 3(-1)^2 = 3,$

$f'(1) = 3(x)^2 = 3$

$\int_{-1}^{1} x^3 dx = \left[\frac{x^4}{4}\right]_{-1}^{1} = 0 = a + b \cdot 1 + c \cdot 3 + d \cdot 3$ — (4)

∴ we have a equations

$a+b = 2$ — (1)
$-a+b+c+d = 0$ — (2)
$a+b-2c+2d = 2/3$ — (3)
$-a+b+3c+3d = 0$ — (4)

(1), (3) $\Rightarrow$ $2-2c+2d = 2/3$
∴ $2(1-c+d) = 2/3$
∴ $1-c+d = 2/3$
$-c+d = 1/3 - 1$

$-c+d = -2/3$
∴ $c-d = 2/3$
∴ $c = 2/3 + d$ — (5)

(1) $\Rightarrow$ $a = 2-b$
∴ (2) $\Rightarrow$ $-(2-b)+b+c+d = 0$
$-2+2b+c+d = 0$
$2b+c+d = 2$

(5) $\Rightarrow$ $2b + 2/3 + d + d = 2$

∴ $2b + 2/3 + 2d = 2$

$b + 1/3 + d = 1$
∴ $b+d = 1 - 1/3$

∴ $b+d = 2/3$ — (6)

(4) $\Rightarrow -(2-b)+b+3(2/3+d)+3d=0$

$\therefore -2+b+b+2+3d+3d=0$

$\therefore -2+2b+6d+2=0$

$\therefore -1+b+3d+1=0$

$\therefore b+3d=0$ — (7)

(6) − (5) $\Rightarrow$

$$\begin{array}{r} b+d=2/3 \\ b+3d=0 \\ - \quad - \\ \hline -2d=2/3 \end{array}$$

$\boxed{d=-1/3}$ — (8)

(5) $\Rightarrow c=2/3+d = 2/3-1/3$

$\therefore \boxed{c=1/3}$ — (9)

(6) $\Rightarrow b+d=2/3$

$\therefore b-1/3=2/3$

$\therefore b=2/3+1/3=3/3$

$\therefore \boxed{b=1}$ — (10)

(1) $\Rightarrow a+b=2$

$\therefore a+1=2$

$\therefore \boxed{a=1}$

$\therefore a=1,\ b=1,\ c=1/3,\ d=-1/3$

Page

Date: / /

Q.1 Derive mid point formula using method of undetermined coefficients.

→ let $n=1$ in method of undetermined coefficients

$\therefore \int_a^b f(x)\,dx \simeq \omega_1 f(x_1)$

It gives exact result when $f(x)$ is of degree $\leq 2\times 1-1 = 1$

$\therefore$ let $f(x)=1$ & $f(x)=x$

for

$f(x)=1 \quad \therefore \int_a^b 1\,dx = \omega_1 \Rightarrow \omega_1 = b-a$ — ①

for

$f(x)=x \quad \therefore \int_a^b x\,dx = \omega_1 x_1 \Rightarrow \omega_1 x_1 = \frac{1}{2}(b^2-a^2)$ — ②

①, ② $\Rightarrow x_1 = \frac{1}{2}(b+a)$

$\therefore$ The resulting gaussian - quadrature formula is

$$\int_a^b f(x)\,dx \simeq (b-a)\, f\left(\frac{a+b}{2}\right)$$

which is same as mid point rule

Assignment 3

Q.1 Define a differential equation. what is ment by soulution of a differential equation ? verify that

i) $y = c_1 e^{x} + c_2 e^{x} x$ is a solution of the differential eq. $y'' - 2y' + y = 0$.

ii) $y = 2\sqrt{c-x}$ is a solution of the differential equation $y + 1/y = 0$.

→ Define a differential eq.

→ A differential equation is an equation invaving independent variable, dependent variable & its one or more derivatives.

ex. $y'' + 4y = 0$

Here y is dependent variable, y is function of x.

→ Solution of differential equation.

→ A solution of a differential equation is a specific function that satisfies the equation.

i) Here diff. eq. is $y'' - 2y' + y = 0$ — (a)

let $y(x) = c_1 e^{x} + c_2 e^{x} x$ — (1)

$$\therefore y'(x) = c_1 e^{x} + c_2 \frac{d}{dx}[x e^{x}]$$

$$= c_1 e^{x} + c_2 \left(x \frac{d}{dx} e^{x} + e^{x} \frac{d}{dx} x\right)$$

$$\left(\because \frac{d}{dx}(uv) = u \frac{d}{dx} v + v \frac{d}{dx} u\right)$$

$$= c_1 e^{x} + c_2 (x e^{x} + e^{x})$$

$$y'(x) = c_1 e^{x} + c_2 e^{x} + c_2 e^{x}$$ — (2)

Scanned with CamScanner

$y''(x) = c_1e^x + c_2e^x + c_2e^x + c_2(e^x + xe^x)$
$y''(x) = c_1e^x + 2c_2e^x + c_2xe^x$ — ③

Subsituting ①, ② & ③ in Ⓔ

$0 = c_1e^x + 2c_2e^x + c_2xe^x - 2(c_1e^x + c_2e^x + c_2xe^x) +$
$c_1e^x + c_2xe^x$

$= 2c_1e^x + 2c_2e^x + 2c_2xe^x - 2c_1e^x - 2c_2e^x - 2c_2xe^x$

$= 0$

$\therefore y = c_1e^x + c_2xe^x$ is a solution of $y'' - 2y' + y = 0$.

ii) Here diff. eq. is $y' + \frac{1}{y} = 0$ —(a) let
$y = 2\sqrt{c-x}$

$y = 2\sqrt{c-x} = 2(c-x)^{1/2}$ — ①
$y' = 2 \times \frac{1}{2}(c-x)^{1/2 - 1} \times \frac{d}{dx}(c-x)$
$= (c-x)^{-1/2} \times -1 = \frac{-1}{\sqrt{c-x}}$ — ②

ⓐ $\Rightarrow \frac{-1}{\sqrt{c-x}} + \frac{1}{2\sqrt{c-x}} \neq 0$

$\therefore y = 2\sqrt{c-x}$ is not a solution of $y' + \frac{1}{y} = 0$

Q2 HOW is ordinary differential equation different from partial differential equation? Give one example of each.

→ when dependent variable is a function of only one variable, then all the derivatives involved in the equation are ordinary & the equation is called ordinary diff eq.

→ If dependent variable is a function of more than one independent variable that is, it is a function of several variables, then the equation contains partial derivatives with respect to different independent variables & the equation is called partial differential Equation.

→ Exp of ordinary differential eq.

$$y'' + 4y = 0$$
$$y' + 1/y = 0$$

→ Example of partial differential eq

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \text{ (laplace's Equation)}$$

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x,y) \text{ (poisson's Equation)}$$

Scanned with CamScanner

Q.3 Differentiate b/w initial value problem & Boundary value problem. classiffy following differential equations in initial value problem & Boundary value problem

→ If the conditions are specified at a single point, these conditions are called initial conditions & the Differential equation combined with conditions is called initial value problem.

→ If the conditions are specified at more then one point. these conditions are called Boundary conditions & the Differential equation combined with conditions is called Boundary value problem.

i) $y' = t - y$, $y(0) = 1$
→ Initial value problem.

ii) $y'' + 9y = e^{x} \sin x$, $y(0) = 1$, $y(1) = 2$
→ Boundary value problem.

iii) $y''' + y = 0$, $y(0) = 0$, $y'(0) = 2$
→ Boundary value problem

iv) $y'' + y' = t^2 + y^2$, $y(0) = 1$, $y'(0) = 1$.
→ Boundary value problem.

v) $y' = t^2 + y^2$; $y(0) = 1$
→ Initial value problem.

Scanned with CamScanner

Q.4 Determine order & degree of following differential eq.

i) $y'' + 4y = e^{x}$

→ order = 2 , degree = 1.

ii) $y'' + 4(y')^3 + y^2 = x^3 + x^2$

order = 2 , degree = 1

iii) $(y''')^2 + (y')^3 + 3y = 5x$

order = 2 , degree = 2

iv) $y' + 2y^2 = x^2$

order = 1 , degree = 1

Q.5 What are the characteristics of Single Step numerical methods to find solutions of first order, first degree IVP?

→ These are the characteristics of Single step numerical methods

→ It is direct

→ It is non iterative.

→ It is Based on Taylor Series method.

→ It uses estimate at previous step & function information.

→ Practically, error can not be estimated.

Scanned with CamScanner

Q.6 Differentiate b/w Single step & multistep numerical methods to obtain Solution of IVP?

Single Step methods	multistep methods
• Direct	More than one Previous step
• Not iterative	. Iterative.
• Based on Taylor series methods	Predictor Corrector formulas.
• Self Starting	Not Self Starting
• Practically, accuracy can not be estimated	practically, accuracy can be estimated.

Q.7 What is Euler formula to solve IVP? Give graphical representation of Euler formula. How is it obtained from Taylor series?

→ Euler formula is a Taylor series of order I. where N = 1. Second & highest Powers of h are ignored.

→ The Taylor formula for solving the IVP $\frac{dy}{dx} = f(x, y(x))$,

$u(x_0) = y_0$, $a \leq x \leq b$, with step size h can be expressed as,

Scanned with CamScanner

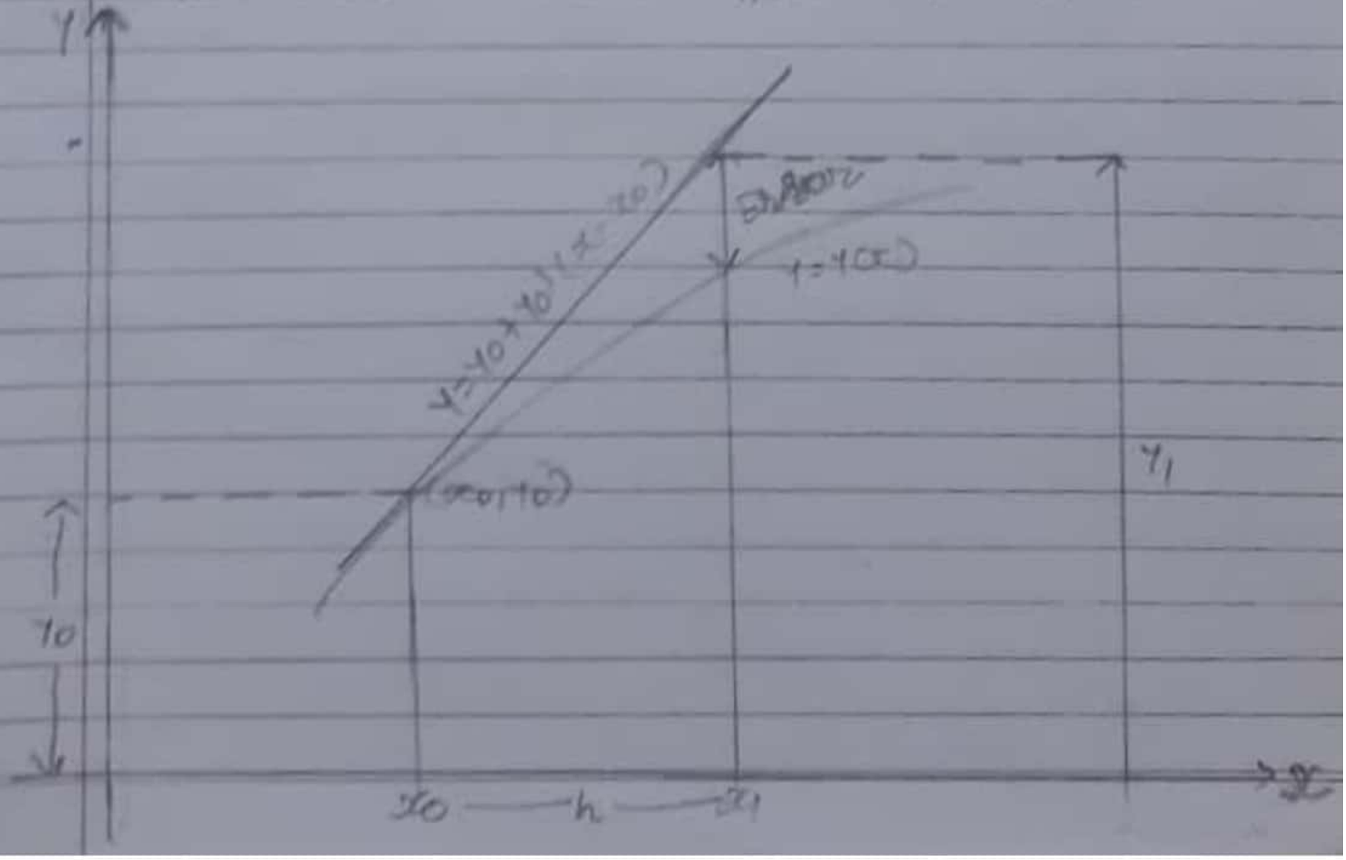
$$y_{i+1} = y_i + h y_i' + O(h^2)$$

→ Ignoring second & higher powers of h, gives $y_{i+1} = y_i + h y_i'$

→ That is same as $y_{i+1} \simeq y_i + h f(x_i, y_i)$

→ Graphical representation of Euler formula
→ slope of tangent to the curve $y(x)$ is $y'(x)$, which is equal to $f(x, y)$. Equation of tangent to the curve $y(x)$ at (x_i, y_i) is $(y - y_i) = y_i'(x - x_i)$
→ this tangent is approximation curve of (x_i, y_i).

(x_{i+1}, y_{i+1}) lies on this line giving
$(y_{i+1} - y_i) = y_i'(x_{i+1} - x_i)$
$\therefore (y_{i+1} - y_i) = h y_i'$ or $y_{i+1} \simeq y_i + h y_i'$



Scanned with CamScanner

Q.8 Apply Euler method to approximate the solution of the given initial value problem over the indicated interval using given stepsize h.

i) $x' = x - t \quad (0 \leq t \leq 1)$, $x(0) = 2$, $h = 0.25$

Here $0 \leq t \leq 1$, $h = 0.25$, $f(x_i, t_i) = x_i - t_i$

$\therefore t_0 = 0$, $t_1 = 0.25$, $t_2 = 0.50$, $t_3 = 0.75$, $t_4 = 1$

at t_0, $x_0 = 2$

Euler formula is,

$$x_{i+1} \cong x_i + h\, f(x_i, t_i)$$

$$\therefore x_{i+1} \cong x_i + h(x_i - t_i)$$

Here $h = 0.25$

$$\therefore x_{i+1} \cong x_i + 0.25(x_i - t_i)$$

$$\therefore x_{i+1} \cong x_i + 0.25x_i - 0.25t_i = 1.25x_i - 0.25t_i$$

at, t_1,

$$x_1 = 1.25(x_0) - 0.25(t_0)$$

$$x_1 = 1.25(2) - 0.25(0) = 1.50 \quad \text{--- (1)}$$

at, t_2,

$$x_2 = 1.25(x_1) - 0.25(t_1)$$

$$x_2 = 1.25(1.5) - 0.25(0.25)$$

$$= 3.125 - 0.0625$$

$$x_2 = 3.0625 \quad \text{--- (2)}$$

at, t_3,

$$x_3 = 1.25(x_2) - 0.25(t_2)$$

$$= 1.25(3.0625) - 0.25(0.50) = 3.82813 - 0.125$$

$$x_3 = 3.70313 \quad \text{--- (3)}$$

Scanned with CamScanner

at t_4,

$$x_4 = 1.25(x_3) - 0.25(t_3)$$
$$= 1.25(3.70313) - 0.25(0.75)$$
$$x_4 = 4.62891 - 0.1875$$
$$= 4.44141 \quad \text{—} \ (4)$$

By Euler method the solution is 4.44141.

(ii) $y' = xy^3 - y \quad (0 \leq x \leq 1), \ y(0)=1, \ h=0.25$

Here $f(x,y) = xy^3 - y$, $0 \leq x \leq 1$, $y(0)=1$, $h=0.25$

$x_0 = 0$. $x_1 = 0.25$, $x_2 = 0.50$, $x_3 = 0.75$, $x_4 = 1$

By Euler formula,

$$y_{i+1} \simeq y_i + h\, f(x_i, y_i)$$
$$= y_i + 0.25(x_i y_i^3 - y_i)$$
$$= y_i + 0.25 x_i y_i^3 - 0.25 y_i$$
$$y_{i+1} = 0.75 y_i + 0.25 x_i y_i^3 \quad \text{—} \ (1)$$

at x_0, $y_0 = 1$.

at x_1

(1) $y_1 = 0.75(y_0) + 0.25(x_0 (y_0)^3)$
$$= 0.75(1) + 0.25(0)(1)^3$$
$$y_1 = 0.75 \quad \text{—} \ (2)$$

Scanned with CamScanner

at x_2.

$y_2 = 0.75(y_1) + 0.25(x_1)(y_1)^3$

$y_2 = 0.75(0.75) + 0.25(0.25)(0.75)^3$

$\therefore y_2 = 0.5625 + 0.02637$

$= 0.58887$ —— (2)

at x_3

$y_3 = 0.75(y_2) + 0.25(x_2)(y_2)^3$

$= 0.75(0.58887) + 0.25(0.50)(0.58887)^3$

$= 0.44163 + 0.02553$

$y_3 = 0.46718$ —— (3)

at x_4

$x_4 = 0.75(y_3) + 0.25(x_3)(y_3)^3$

$= 0.75(0.46718) + 0.25(0.75)(0.46718)^3$

$= 0.35039 + 0.01912$

$x_4 = 0.3695$

$\therefore$ Solution $= 0.3695$

iii) $y' = \cos 2t - \sin 3t, \quad 0 \leq t \leq 1, \; y(0) = 1, \; h = 0.2$

$\rightarrow f(t, y) = \cos 2t - \sin 3t, \quad 0 \leq t \leq 1, \; y(0) = 1, \; h = 0.2$

$\therefore t_0 = 0, \; t_1 = 0.2, \; t_2 = 0.4, \; t_3 = 0.6, \; t_4 = 0.8, \; t_5 = 1$

By Euler formula.

$y_{i+1} \cong y_i + h \, f(t_i, y_i)$

$\cong y_i + 0.2(\cos 2t_i - \sin 3t_i)$ —— (1)

at t_0: $y_0 = 1$.

Scanned with CamScanner

at t_1,

1) $y_1 = y_0 + 0.2(\cos 2t_0 - \sin 3t_0)$
$= 1 + 0.2(\cos 0 - \sin 0)$
$y_1 = 1 + 0.2(1-0) = 1 + 0.25 = 1.2$ — (2)

at t_2,

$y_2 = y_1 + 0.2(\cos 2t_1 + \sin 3t_1)$
$= 1.2 + 0.2(\cos 0.4 + \sin 0.6)$
$= 1.2 + 0.2(0.92106 + 0.56464)$
$y_2 = 1.49714$ — (3)

at t_3,

$y_3 = y_2 + 0.2(\cos 2t_2 + \sin 3t_2)$
$= 1.49714 + 0.2(\cos 0.8 + \sin 1.2)$
$y_3 = 1.82289$ — (4)

at t_4,

$y_4 = y_3 + 0.2(\cos 2t_3 + \sin 3t_3)$
$= 1.82289 + 0.2(\cos 1.2 + \sin 1.8)$
$= 2.09013$ — (5)

at t_5,

$y_5 = y_4 + 0.2(\cos 2t_4 + \sin 3t_4)$
$= 2.09013 + 0.2(\cos 1.6 + \sin 2.4)$
$= 2.21939$ — (6)

$\therefore$ Solution is $\underline{\underline{2.21939}}$

Scanned with CamScanner

Assignment - 4

Ordinary Differential Equations-II

Page : 41
Date: / /

Q.1 A ball at 1200 k is allowed to cool down in air at an ambient temprature of 300k. Assuming heat is lost only due to radiation, the differential equation for temprature of the ball is given by $d\theta/dt = -2.2067 \times 10^{-12} (\theta^4 - 81 \times 10^8)$, $\theta(0) = 1200k$

where θ is in k & t in seconds. Find the temperature at t = 480 seconds using Runge kutta 2nd order Taylor's method. Assume a step size of h = 240 seconds.

→ Runge kutta formulas of order 2 is

$dy/dx = f(x, y(x))$, $y(x_0) = y_0$, $a \leq x \leq b$

$y_{i+1} = y_i + 1/2 (k_0 + k_1)$ with

$k_0 = hf(x_i, y_i)$
$k_1 = hf(x_i + h, y_i + k_0)$ for $i = 0, 1, 2, \ldots, n$

→ Here we have $d\theta/dt = f(t, \theta(t)) = -2.2067 \times 10^{-12} (\theta^4 - 81 \times 10^8)$, $t_0 = 0$

$\theta_0 = \theta(0) = 1200k$, $0 \leq t \leq 480$, $h = 240 sec$
let $t_1 = 240$, $t_2 = 480$
$\therefore \theta_1 = \theta_0 + 1/2 (k_0 + k_1)$ —— (1)

$k_0 = 240 * f(t_0, \theta_0) = 240 \times f(0, 1200)$
$= 240 \times (-2.2067 \times 10^{-12} \times ((1200)^4 - 81 \times 10^8))$
$k_0 = -1093.9033$ —— (2)

Scanned with CamScanner

$k_1 = h \times f(t_0+h, \theta_0+k_0) = 240 \times f(240, 1200+(-1093.9053))$

$= 240 \times f(240, 106.104)$

$= 240 \times (-2.2067 \times 10^{-12}(106.104)^4 - 81 \times 10^8)$

$k_1 = 4.2227$ —— (3)

Substituting eq. (2) & (3) in (1),

$\theta_1 = 1200 + 1/2(-1093.9053 + 4.2227)$

$\theta_1 = 655.1587$ —— (4)

→ Now $t_1 = 480$ sec

$\theta_2 = \theta_1 = 1/2(k_0 + k_1)$

$k_0 = hf(t_1, \theta_1) = 240 \times f(240, 655.1587)$

$= 240 \times -2.2067 \times 10^{-12}(655.1587^4 - 81 \times 10^8)$

$k_0 = -93.2856$

$k_1 = hf(t_1+h, \theta_1+k_0) = 240 \times f(480, (655.1587 - 93.2856))$

$k_1 = 240 \times f(480, 561.8731)$

$= 240 \times (-2.2067 \times 10^{-12}(561.8731^4 - 81 \times 10^8)$

$k_1 = -48.4948$

$\theta_2 = 655.1587 + 1/2(-93.2856 - 48.4948)$

$\theta_2 = 584.2685\,K$ —— (5)

Q.2 Use Taylor's method of order 2 to approximate the solutions of each of the following problems

(a) $dy/dx = y - x^2 + 1$, $y(0) = 0.5$, $0 \leq x \leq 2$, $h = 0.5$

→ Here $dy/dx = y' = y - x^2 + 1$, $x_0 = 0$, $y_0 = 0.5$, $0 \leq x \leq 2$, $h = 0.5$

Scanned with CamScanner

Scanned with CamScanner

Page: 43
Date: / /

at $x_i = 0.5$, By Taylor's method of order 2

$$y_{i+1} \cong y_i + h y_i' + \frac{h^2}{2!} y_i''$$

Here $y' = y - x^2 + 1 \therefore y'' = \frac{dy'}{dx} = \frac{d}{dx}(y - x^2 + 1) = -2x$

$$\therefore y_1 = y_0 + h y_0' + \frac{h^2}{2!} y_0''$$

$$= 0.5 + 0.5(y_0 - x_0^2 + 1) + \frac{(0.5)^2}{2!}(-2x_0)$$

$$= 0.5 + 0.5(0.5 - 0 + 1) + \frac{(0.5)^2}{2!}(-2 \times 0)$$

$$y_1 = 1.25 \quad \text{— (1)}$$

→ at $x_2 = 1$

$$y_2 = y_1 + h y_1' + \frac{h^2}{2!} y_1''$$

$$= 1.25 + 0.5(y_1 - x_1^2 + 1) + \frac{(0.5)^2}{2!}(-2x_1)$$

$$= 1.25 + 0.5(1.25 - (0.5)^2 + 1) + 0.25/2\,(-2.(0.5))$$

$$y_2 = 2.125 \quad \text{— (2)}$$

→ at $x_3 = 1.50$

$$y_3 = y_2 + h y_2' + \frac{h^2}{2!} y_2''$$

$$= 2.125 + 0.5(y_2 - x_2^2 + 1) + \frac{(0.5)^2}{2!}(-2x_2)$$

$$= 2.125 + 0.5(2.125 - 1^2 + 1) + 0.25/2\,(-2(1))$$

$$y_3 = 2.3375 \quad \text{— (3)}$$

→ at $x_4 = 1$

$$y_4 = y_3 + h y_3' + \frac{h^2}{2!} y_3''$$

Scanned with CamScanner

$= 2.9375 + 0.5(2.9375 - (1.5)^2 + 1) + 0.25/2(-2(1.5))$

$y_4 = 3.40625$

b) $dy/dt = \cos 2t + \sin 3t, \ 0 \le t \le 1, \ y(0) = 1, \ h = 0.25.$

→ Here $dy/dt = y' = \cos 2t + \sin 3t, \ 0 \le t \le 1$

$y_0 = 1, \ t_0 = 0, \ h = 0.25, \ y' = \cos 2t + \sin 3t$

$y'' = -2\sin 2t + 3\cos 3t$

→ at $t_1 = 0.25$, using taylor's method of order 2

$y_1 = y_0 + hy'_0 + h^2/2! \, y''_0$

$= 1 + 0.25(\cos 2t_0 + \sin 3t_0) + \frac{(0.25)^2}{2!}(-2\sin 2t_0 + 3\cos 3t_0)$

$= 1 + 0.25(\cos 0 + \sin 0) + \frac{0.0625}{2}(-2\sin 0 + 3\cos 0)$

$y_1 = 1.34375$ —— ①

at $t_2 = 0.5$

$y_2 = y_1 + hy'_1 + h^2/2! \, y''_2$

$= 1.34375 + 0.25(\cos 2t_1 + \sin 3t_2) + \frac{(0.25)^2}{2!}(-2\sin 2t_1 + 3\cos 3t_1)$

$= 1.34375 + 0.25(\cos(0.5) + \sin(0.75)) + \frac{0.0625}{2}$

$(-2\sin \cos 5) + 3\cos(0.75))$

$y_2 = 1.7722$ —— ②

Scanned with CamScanner

→ at $t_3 = 0.75$

$$y_3 = 1.7722 + (0.27)(\cos 1 + \sin 1.5) + \frac{0.0625}{2}(-2\sin 1 + 3\cos(1.5))$$

$$y_3 = 2.1107 \quad \text{--- (3)}$$

→ at $t_4 = 1$

$$y_4 = 2.1107 + (0.25)(\cos(1.5) + \sin(2.25)) + \frac{(0.0625)}{2}(-2\sin(1.5) + 3\cos(2.25))$$

$$y_4 = 2.20167$$

c) $y' = \dfrac{1+t}{1+y} \quad 1 \le t \le 2, \; y(1) = 2, \; h = 0.5$

Here, $t_0 = 1$, $y_0 = 2$, $1 \le t \le 2$, $h = 0.5$

$$y' = \frac{1+t}{1+y}, \quad y'' = \frac{1}{1+y}\frac{d}{dt}(1+t) = \frac{1}{1+y}$$

Using Taylor's method of order 2,

at $t_1 = 1.5$

$$y_1 = y_0 + h y_0' + \frac{h^2}{2!} y_0''$$

$$= 2 + 0.5\left(\frac{1+t_0}{1+y_0}\right) + \frac{(0.5)^2}{2}\left(\frac{1}{1+y_0}\right)$$

$$= 2 + 0.5\left(\frac{1+1}{1+2}\right) + \frac{0.25}{2}\left(\frac{1}{1+2}\right)$$

$$y_1 = 2.375$$

Scanned with CamScanner

→ at $t_2 = 2$.

$y_2 = y_1 + h y_1' + \frac{h^2}{2!} y_1''$

$= 2.375 + (0.5)\left(\frac{1+t_1}{1+y_1}\right) + \frac{0.25}{2}\left(\frac{1}{1+y_1}\right)$

$= 2.375 + 0.5\left(\frac{1+1.5}{1+2.375}\right) + 0.125\left(\frac{1}{1+2.375}\right)$

$\underline{y_2 = 2.7824}$

d) $y' = 1 + y/t$, $y(1) = 1$, $h = 0.25$, $1 \le t \le 1.5$

Here $t_0 = 1$, $y_0 = 1$, $h = 0.25$, $1 \le t \le 1.5$

$y' = 1 + y/t$, $y'' = y \frac{d}{dt}\left(\frac{1}{t}\right) = y \frac{d}{dt}(t^{-1})$

$\therefore y'' = -y/t^2$

Using Taylor's method of order 2.

at $t_1 = 1.25$.

$y_1 = y_0 + h y_0' + h^2/2!\, y_0''$

$= 1 + 0.25(1 + 1/1) + 0.0625/2\, (-1/1)$

$y_1 = 1.46875$ —— ①

→ at $t_2 = 1.50$

$y_2 = y_1 + h y_1' + h^2/2!\, y_1''$

$= 1.46875 + 0.25 \times \left(1 + \frac{1.46875}{1.25}\right) + 0.03125\left(\frac{-1.46875}{(1.25)^2}\right)$

$y_2 = \boxed{1.983125}$ —— ②

Scanned with CamScanner

e) $y' = xe^{y+x} - 1$, $y(0) = -1$, $h = 0.5$, $0 \le x \le 1$

Here, $y_0 = -1$, $x_0 = 0$, $h = 0.5$, $0 \le x \le 1$

$y' = xe^{y+x} - 1$, $y'' = x \frac{d}{dx}(e^{y+x}) + e^{y+x} \frac{d}{dx} x$

$y'' = xe^{y+x} + e^{y+x}$

Using Taylor's method of order 2.

at $x_1 = 0.5$.

$y_1 = y_0 + hy_0' + h^2/2!\, y_0''$

$= -1 + 0.5(0 \times e^{-1+0} - 1) + \frac{0.25}{2}(0 \times e^{-1+0} + e^{-1+0})$

$y_1 = -1.4540$ — (1)

→ at $x_2 = 1$

⇒ $y_2 = y_1 + hy_1' + h^2/2!\, y_1''$

$= -1.4540 + 0.5(0.5 \times e^{(-1.4540+0.5)} - 1) + \frac{0.25}{2}$
$[0.5\, e^{(-1.4540+0.5)} + e^{(-1.4540+0.5)})$

$y_2 = -1.7855$

Q.3 Compare the Performance of Euler's method, Second order Taylor's method, RK of order 2 & RK order 4 for the following differential eq.

Scanned with CamScanner

Page: 4.8
Date: / /

$\frac{dy}{dx} = x/y, \quad 0 \le x \le 2, \quad y(0) = 1, \quad h = 0.5$

Here $dy/dx = x/y$, $h = 0.5$, $y_0 = 1$, $x_0 = 0$, $0 \le x \le 2$

1) with Euler's method:

$y_{i+1} = y_i + hf(x_i, y_i)$

$y_{i+1} = y_i + 0.5(x_i/y_i)$

→ at $x_0 = 0$, $y_0 = 1$

→ at $x_1 = 0.5$

$y_1 = y_0 + 0.5(x_0/y_0) = 1 + 0.5(0/1) = 1$

→ at $x_2 = 1$

$y_2 = y_1 + 0.5(x_1/y_1) = 1 + 0.5(0.5/1) = 1.25$

→ at $x_3 = 1.5$

$y_3 = y_2 + 0.5(x_2/y_2) = 1.25 + 0.5(1/1.25) = 1.65$

→ at $x_4 = 2$.

$y_4 = y_3 + 0.5(x_3/y_3) = 1.65 + 0.5(1.5/1.65) = 2.1045$

⇒ Second order Taylor's method:

Here $y' = x/y$, $y'' = 1/y$, $x_0 = 0$, $y_0 = 1$, $h = 0.5$, $0 \le x \le 2$.

$y_{i+1} = y_i + hy_i' + h^2/2!\; y_i''$

→ at $x_1 = 0.5$

$y_1 = y_0 + h(x_0/y_0) + h^2/2!\,(1/y_0) = 1 + 0.5(0/1) + \frac{(0.5)^2}{2!}(1/1)$

$= 1.125$

→ at $x_2 = 1$

$y_2 = 1.125 + 0.5\left(\frac{0.5}{1.125}\right) + 0.125\left(\frac{1}{1.125}\right) = 1.4583$

→ at $x_3 = 1.5$

$y_3 = 1.4583 + 0.5\left(\frac{1}{1.4583}\right) + 0.125\left(\frac{1}{1.4583}\right) = 1.8869$

→ at $x_4 = 2$

$y_4 = 1.8869 + 0.5\left(\frac{1.50}{1.8869}\right) + 0.125\left(\frac{1}{1.8869}\right) = 2.3506$

⇒ ③ RK of order 2

$y_{i+1} = y_i + \frac{1}{2}(k_0 + k_1)$

$k_0 = hf(x_i, y_i)$, $k_1 = hf(x_i + h ; y_i + k)$

→ at $x_1 = 0.5$

$y_1 = y_0 + \frac{1}{2}(k_0 + k_1)$

$k_0 = 0.5(x_0 / y_0) = 0.5\left(\frac{0}{1}\right) = 0.$

$k_1 = 0.5 f(x_0 + 0.5, y_0 + 0) = 0.5 f(0.5, 1)$

$k_1 = 0.5\left(\frac{0.5}{1}\right) = 0.05$

$y_1 = 1 + \frac{1}{2}(0 + 0.25) = 1.125$

→ at $x_2 = 1$

$k_0 = 0.5(0.5/1.125) = 0.2$

$k_1 = 0.5 f(0.5 + 0.5, 1.125 + 0.2) = 0.5 f(1, 1.325)$

$k_1 = 0.3774$

$y_2 = 1.125 + \frac{1}{2}(0.2 + 0.3774) = 1.4157$

→ at $x_3 = 1.5$

$k_0 = 0.5\left(\frac{1}{1.4137}\right) = 0.3537$

$k_1 = 0.5 f(1.5, 1.7874) = 0.4244$

$y_3 = 1.4137 + \frac{1}{2}(0.3537 + 0.4244) = 1.80275$

Scanned with CamScanner

→ at $x_4 = 2$

$k_0 = 0.5\,(^{1.5}/1.8028) = 0.4162$

$k_1 = 0.5\,f(2, 2.1565) = 0.4637$

$y_4 = 1.8028 + 1/2\,(0.4162 + 0.4037) = 2.2428$

→ (4) RK order 4.

$y_{i+1} = y_i + 1/6\,(k_0 + 2k_1 + 2k_2 + k_3)$

$k_0 = hf(x_i, y_i)$, $k_1 = hf(x_i + h/2, y_i + k_0/2)$

$k_2 = hf(x_i + h/2, y_i + k_1/2)$

$k_3 = hf(x_i + h, y_i + k_2)$

→ at $x_1 = 0.5$

$k_0 = 0.5\,(0/1) = 0$

$k_1 = 0.5\,f(0.5/2, 1) = 0.125$

$k_2 = 0.5\,f(0.5/2, 1 + 0.125/2) = 0.5\,f(0.25, 1.0625) = 0.1176$

$k_3 = 0.5\,f(0.5, 1 + 1.1176) = 0.5\,f(0.5, 2.1176) = 0.1181$

$y_1 = 1 + 1/6\,(0 + 2(0.125) + 2(0.1176) + 0.1181)$

$y_1 = 1.10055$

→ at $x_2 = 1$

$k_0 = 0.5\,(^{0.5}/1.10055) = 0.2272$

$k_1 = 0.5\,f(0.75, 1.21415) = 0.3089$

$k_2 = 0.5\,f(0.75, 1.255) = 0.2989$

$k_3 = 0.5\,f(1, 1.39945) = 0.3573$

$y_2 = 1 + 1/6\,(0.2272 + 2(0.3089) + 2(0.2989) + 0.3573)$

$y_2 = 1.3$

Scanned with CamScanner

→ at $x_3 = 1.5$

$k_0 = 0.5(1/1.3) = 1.2692.$
$k_1 = 0.5(1.25/1.9846) = 0.3931.$
$k_2 = 0.5(1.25/1.4655) = 0.4276$
$k_3 = 0.5(1.5/1.7276) = 0.4341$
$y_3 = 1.3 + 1/6(1.2692 + 2(0.3931) + 2(0.4276) + 0.4341)$

$y_3 = 1.8341$

→ at $x_4 = 2.$

$k_0 = 0.5(1.5/1.8341) = 0.4089$
$k_1 = 0.5(1.75/2.0386) = 0.3554$
$k_2 = 0.5(1.75/2.2633) = 0.3866$
$k_3 = 0.5(2/2.2207) = 0.4503$
$y_4 = 1.8341 + 1/6(0.4089 + 2(0.3554) + 2(0.3866) + 0.4503)$

$y_4 = 2.3723.$

→ Table of Comparison
for $dy/dx = x/y$, $h = 0.5$, $0 \le x \le 2.$

x	Euler	Taylor order 2	RK2	RK4	Exact sol.
0	1	1	1	1	1
0.5	1	1.125	1.125	1.10053	1.1180
1	1.25	1.4523	1.4137	1.9	1.4142
1.5	1.65	1.8869	1.8048	1.8341	1.8028
2	2.1045	2.3506	2.2428	2.3723	2.2361

Scanned with CamScanner

Q.4 Explain Taylor's method of order n to approximate a solution to differential equation $y'(x) = f(x, y(x))$ $a \le x \le b$, $y(a) = c$. What are its advantages? Why is it seldom used in practice for computer implementation?

→ The Taylor's method of order n is

$$y(x) = y_i + (x-x_i)y_i' + \frac{(x-x_i)^2}{2!}y_i'' + \frac{(x-x_i)^3}{3!}y_i''' + \dots + \frac{(x-x_i)^n}{n!}y_i^{(n)}$$

→ Using this taylor Series to evaluate $y'(x) = f(x, y(x))$, $a \le x \le b$, $y(a) = c$

$$y_{i+1} = y_i + hy_i' + \frac{h^2}{2!}y_i'' + \dots + \frac{h^n}{n!}y_i^{(n)}$$

where $y^{(n)}$ is for n^{th} derivative of $y(x)$.
at $x = x_i$

→ Taylor's Series is seldom used in practical for Computer implementation due to requirement of derivation of expression of derivatives & evaluation of their values.

→ Taylor's Series is back bone of direct single step formulas. Euler formula & family of Runge kutta formulas are derived from Taylor series method.

Scanned with CamScanner

Page: 413
Date: / /

Q.5 Repeat Question 2 with Runge kutta method of order 2.

a) $dy/dx = y - x^2 + 1$, $y(0) = 0.5$, $0 \le x \le 2$, $h = 0.5$

→ Here, $dy/dx = y - x^2 + 1$, $y(0) = 0.5$, $0 \le x \le 2$, $h = 0.5$

$y_0 = 0.5$, $x_0 = 0$

→ RK method of order 2 is

$y_{i+1} = y_i + 1/2 \, (k_0 + k_1)$

$k_0 = hf[x_i, y_i]$, $k_1 = hf(x_i + h, y_i + k_0)$

→ at $x_1 = 0.5$

$y_1 = y_0 + 1/2 \, (k_0 + k_1)$

$k_0 = 0.5 \times f(x_0, y_0) = 0.5 \times (0.5 - 0^2 + 1)$

$k_0 = 0.75$

$k_1 = 0.5 \times f(x_0 + 0.5, y_0 + 0.75) = 0.5 \times f(0.5, 1.25)$

$k_1 = 0.5(1.25 - 0.5^2 + 1) = 1$

$y_1 = 0.5 + 1/2 \, (0.75 + 1) = 1.375$

→ at $x_2 = 1$

$k_0 = 0.5(1.375 - (0.5)^2 + 1) = 1.0625$

$k_1 = 0.5f(0.5 + 0.5, 1.375 + 1.0625) = 0.5f(1, 2.4375)$

$k_1 = 1.2188$

$y_2 = 1.375 + 1/2 \, (1.0625 + 1.2188) = 2.5156$

→ at $x_3 = 1.5$

$k_0 = 0.5(2.5156) = 1.2578$

$k_1 = 0.5 \times (2.5234) = 1.2617$

$y_3 = 2.5156 + 1/2 \, (1.2578 + 1.2617) = 3.7754$

Scanned with CamScanner

→ at $x_4 = 2$.

$k_0 = 1.2627$

$k_1 = 1.0192$

$y_4 = 4.9163$

b) $dy/dt = \cos 2t + \sin 3t$, $0 \le t \le 1$, $y(0) = 1$, $h = 0.25$

→ RK method of order 2 is

$y_{i+1} = y_i + \frac{1}{2}(k_0 + k_1)$

$k_0 = hf(x_i, y_i) = hf(t_i, y_i)$

$k_1 = hf(x_i + h, y_i + k_0) = hf(t_i + h, y_i + k_0)$

→ at point $t_0 = 0$, $y_0 = 1$.

→ at point $t_1 = 0.25$

$k_0 = 0.25 f(0,1) = 0.25(1) = 0.25$

$k_1 = 0.25 f(0 + 0.25, 1 + 0.25) = 0.25 \times (\cos(2 \times 0.25) + \sin(3 \times 0.25))$

$k_1 = 0.3898$

$y_1 = 1 + 1/2(0.25 + 0.3898)$

$y_1 = 1.3199$

→ at Point $t_2 = 0.50$

$k_0 = 0.25 f(0.25, 1.3199) = 0.3898$

$k_1 = 0.25 f(\cos(2 \times 0.50) \times \sin 3(0.50))$

$k_1 = 0.3845$

$y_2 = 1.3199 + \frac{1}{2}(0.3898 + 0.3845) = 1.7071$

→ at Point $t_3 = 0.75$.

$k_0 = 0.25 \times (\cos 2(0.50) + \sin 3(0.50))$
$k_0 = 0.3845$.
$k_1 = 0.25 \times (\cos(2\times0.75) + \sin(3\times0.75))$
$k_1 = 0.2122$
$y_3 = 1.7071 + \frac{1}{2}(0.3845 + 0.2122) = 2.0055$

→ at point $t_4 = 1$

$k_0 = 0.25 \times (\cos(2\times0.75) + \sin(3\times0.75))$
$k_0 = 0.2122$.
$k_1 = 0.25 \times (\cos(2\times1) + \sin(3\times1))$
$k_1 = -0.0658$
$y_4 = 2.0055 + \frac{1}{2}(0.2122 - 0.0658)$
$y_4 = 2.0772$

c) $y' = \frac{1+t}{1+y}$, $1 \le t \le 2$, $y(1) = 2$, $h = 0.5$

RK method of order 2.

$y_{i+1} = y_i + \frac{1}{2}(k_0 + k_1)$
$k_0 = hf(t_i, y_i)$
$k_1 = hf(t_i + h, y_i + k_0)$

→ at Point $t_0 = 1$, $y_0 = 2$

→ at point $t_1 = 1.5$

$k_0 = 0.5 f(1,2) = 0.5 \times \left(\frac{1+1}{1+2}\right) = 0.3$
$k_1 = 0.5 \times f(1+1.5, 2+0.3) = 0.378$
$y_1 = 2 + \frac{1}{2}(0.3 + 0.378) = 2.339$

Scanned with CamScanner

→ at Point $t_2 = 2$.

$k_0 = 0.5 f(1.5, 2.339) = 0.3744$

$k_1 = 0.5 f(2, 2.339 + 0.3744) = 0.4039$.

$y_2 = 2.339 + \frac{1}{2}(0.3744 + 0.4039)$

$y_2 = 2.7282$

d) $y' = 1 + y/t$, $y(1) = 1$, $h = 0.25$, $1 \leq t \leq 1.5$

RK method of order 2.

$y_{i+1} = y_i + \frac{1}{2}(k_0 + k_1)$

$k_0 = hf(t_i, y_i)$

$k_1 = hf(t_i + h, y_i + k_0)$

→ At point $t_0 = 1$, $y_0 = 1$

→ At Point $t_1 = 1.25$

$k_0 = 0.25 f(1, 1) = 0.25 \times (1 + 1/1) = 0.5$

$k_1 = 0.25 f(1.25, 1.5) = 0.55$

$y_1 = 1 + \frac{1}{2}(0.5 + 0.55) = 1.525$

→ At Point $t_2 = 1.5$

$k_0 = 0.25 f(1.25, 1.525) = 0.25 \times \left(1 + \frac{1.525}{1.25}\right) = 0.555$

$k_1 = 0.25 f(1.25 + 0.25, 1.525 + 0.555)$

$k_1 = 0.25 f(1.5, 2.08)$

$k_1 = 0.25 \times (1 + 2.08/1.5)$

$k_1 = 0.596$

Scanned with CamScanner

$y_2 = 1.525 + \frac{1}{2}(0.555 + 0.596)$

$y_2 = 2.1005$

c) $y' = xe^{y+x} - 1$, $y(0) = -1$, $0 \leq x \leq 1$, $h = 0.5$

R.K method of order 2 is

$y_{i+1} = \frac{1}{2} y_i + \frac{1}{2}(k_0 + k_1)$

$k_0 = hf(x_i, y_i)$

$k_1 = hf(x_i + h, y_i + k_0)$

→ At Point $x_0 = 0$, $y_0 = -1$

→ At Point $x_1 = 0.5$

$k_0 = 0.5 f(0, -1) = 0.5(0e^{-1+0} - 1) = -0.5$

$k_1 = 0.5 f(0+0.5, -1+(-0.5)) = 0.5 f(0.5, -1.5)$

$k_1 = 0.5 \times (0.5 \times e^{-1.5+0.5} - 1) = -0.4080$

$y_1 = -1 + \frac{1}{2}(-0.5 - 0.4080) = -1.454$

→ At Point $x_2 = 1$

$k_0 = 0.5 f(0.5, -1.454) = 0.5(0.5 \times e^{-1.454+0.5} - 1)$

$k_0 = -0.4037$

$k_1 = 0.5 f(0.5+0.5, -1.454 - 0.4037)$

$k_1 = 0.5 f(1, -1.8577)$

$k_1 = 0.5 \times (1 \times e^{-1.8577+1} - 1)$

$k_1 = -0.2879$

$y_2 = -1.454 + \frac{1}{2}(-0.4037 - 0.2879)$

$y_2 = -1.7998$

Scanned with CamScanner

Q.6 Repeat Question 2 with Runge Kutta method of order 4.

a) $dy/dx = y - x^2 + 1$, $y(0) = 0.5$, $0 \le x \le 2$, $h = 0.5$

R.K method of order 4 is

$y_{i+1} = y_i + 1/6 (k_0 + 2k_1 + 2k_2 + k_3)$

$k_0 = hf(x_i, y_i)$

$k_1 = hf(x_i + h/2, y_i + k_0/2)$

$k_2 = hf(x_i + h/2, y_i + k_1/2)$

$k_3 = hf(x_i + h, y_i + k_2)$

→ Here $f(x, y) = y - x^2 + 1$, $y(0) = 0.5$, $0 \le x \le 2$, $h = 0.5$

→ at point $x_0 = 0$, $y_0 = 0.5$

→ at point $x_1 = 0.5$

$k_0 = 0.5 f(0, 0.5) = 0.5 (0.5 + 0^2 + 1) = 0.75$

$k_1 = 0.5 f(0 + 0.5/2, 0.5 + 0.75/2) = 0.5 f(0.25, 0.875)$

$k_1 = 0.90625$

$k_2 = 0.5 f(0.25, 0.5 + 0.90625/2) = 0.5 f(0.25, 0.9531)$

$k_2 = 0.9453$

$k_3 = 0.5 f(0.5, 0.5 + 0.9453) = 0.5 f(0.5, 1.4453)$

$k_3 = 1.09765$

$y_1 = 0.5 + 1/6 (0.75 + 2(0.90625) + 2 \times (0.9453) + 1.09765)$

$y_1 = 1.4251$

Scanned with CamScanner

→ At $x_2 = 1$

$k_0 = 0.5f(0.5, 1.4251) = 1.08755$

$k_1 = 0.5f(0.75, 1.9689) = 1.2032$

$k_2 = 0.5f(0.75, 2.0267) = 1.2321$

$k_3 = 0.5f(1, 2.6572) = 1.3286$

$y_2 = 1.4251 + \frac{1}{6}(1.08755 + 2(1.2032) + 2 \times (1.2321) + 1.3286)$

$y_2 = 2.6396$

→ At Point $x_3 = 1.5$

$k_0 = 0.5f(1, 2.6396) = 1.3198$

$k_1 = 0.5f(1.25, 3.2995) = 1.3685$

$k_2 = 0.5f(1.25, 3.32385) = 1.3807$

$k_3 = 0.5f(1.5, 4.0203) = 1.3852$

$y_3 = 2.6396 + \frac{1}{6}[1.3198 + 2(1.3685) + 2(1.3807) + 1.3852)$

$y_3 = 4.0068$

→ At point $x_4 = 2$

$k_0 = 0.5f(1.5, 4.0068) = 1.3784$

$k_1 = 0.5f(1.75, 4.696) = 1.31675$

$k_2 = 0.5f(1.75, 4.6652) = 1.3014$

$k_3 = 0.5f(2, 5.3082) = 1.1541$

$y_4 = 4.0068 + \frac{1}{6}[1.3784 + 2(1.3167) + 2(1.3014) + 1.1541)$

$y_4 = 5.3016$

Scanned with CamScanner

Page: 4.20.
Date: / /

b) $dy/dx = \cos 2t + \sin 3t$, $0 \leq t \leq 1$, $y(0) = 1$, $h = 0.25$

→ At point $t_0 = 0$, $y_0 = 1$

→ At Point $t_1 = 0.25$

$k_0 = 0.25 f(0, 1) = 0.25$
$k_1 = 0.25 f(0.125, 1.125) = 0.3338$
$k_2 = 0.25 f(0.125, 1.1669) = 0.3338$
$k_3 = 0.25 f(0.25, 1.3338) = 0.3898$

$y_1 = 1 + 1/6 (0.25 + 2(0.3338) + 2(0.3338) + 0.3892$
$y_1 = 1.3292$

→ At point $t_2 = 0.50$

$k_0 = 0.25 f(0.25, 1.3292) = 0.3898$
$k_1 = 0.25 f(0.375, 1.5241) = 0.4085$
$k_2 = 0.25 f(0.375, 1.5334) = 0.4085$
$k_3 = 0.25 f(0.50, 1.7377) = 0.3845$

$y_2 = 1.3292 + 1/6 (0.3898 + 2(0.4085) + 2(0.4085) + 0.3845)$
$y_2 = 1.7306$

→ At Point $t_3 = 0.75$

$k_0 = 0.25 f(0.50, 1.7306) = 0.3845$
$k_1 = 0.25 f(0.625, 1.92285) = 0.3174$
$k_2 = 0.25 f(0.625, 1.8893) = 0.3174$
$k_3 = 0.25 f(0.75, 2.048) = 0.2122$

$y_3 = 1.7306 + 1/6 (0.3854 + 2(0.3174) + 2(0.3174) + 0.2122)$

$y_3 = 2.0428$

Scanned with CamScanner

→ At Point $t_4 = 1$

$k_0 = 0.25\, f(0.75, 2.0418) = 0.2122$

$k_1 = 0.25\, f(0.875, 2.1479) = 0.0789$

$k_2 = 0.25\, f(0.875, 2.0813) = 0.0789$

$k_3 = 0.25\, f(1, 2.1207) = -0.0688$

$y_4 = 2.0428 + \frac{1}{6}[0.2122 + 4(0.0789) + (-0.0688)]$

$y_4 = \underline{2.1183}$

C) $y' = \frac{1+t}{1+y}$, $1 \le t \le 2$, $y(1) = 2$, $h = 0.5$

At $t_0 = 1$, $y_0 = 2$

→ At $t_1 = 1.5$

$k_0 = 0.5\, f(1, 2) = 0.3333$

$k_1 = 0.5\, f(1.25, 2.1667) = 0.3553$

$k_2 = 0.5\, f(1.25, 2.1777) = 0.3540$

$k_3 = 0.5\, f(1.5, 2.354) = 0.3727$

$y_1 = 2 + \frac{1}{6}[0.3333 + 2(0.3553) + 2(0.3540) + 0.3727]$

$y_1 = 2.3541$

→ At $t_2 = 2$

$k_0 = 0.5\, f(1.5, 2.3541) = 0.3727$

$k_1 = 0.5\, f(1.75, 2.5405) = 0.38836$

$k_2 = 0.5\, f(1.75, 2.5483) = 0.3875$

$k_3 = 0.5\, f(2, 2.7416) = 0.40090$

$y_2 = 2.3541 + \frac{1}{6}(0.3727 + 2(0.3884) + 2(0.3875) + 0.4009)$

$y_2 = \underline{2.7417}$

Scanned with CamScanner

d) $y' = 1 + y/t$, $y(1) = 1$, $1 \le t \le 1.5$, $h = 0.25$

At $t_0 = 1$, $y_0 = 1$.

→ At $t_1 = 1.25$

$k_0 = 0.25 f(1, 1) = 0.5$
$k_1 = 0.25 f(1.125, 1.25) = 0.5278$
$k_2 = 0.25 f(1.125, 1.2639) = 0.5309$
$k_3 = 0.25 f(1.25, 1.5309) = 0.5562$

$y_1 = 1 + 1/6 (0.5 + 2(0.5278) + 2(0.5309) + 0.5562)$
$y_1 = 1.5289$

→ At $t_2 = 1.5$

$k_0 = 0.25 f(1.25, 1.5289) = 0.5558$
$k_1 = 0.25 f(1.375, 1.8068) = 0.5785$
$k_2 = 0.25 f(1.375, 1.8182) = 0.5806$
$k_3 = 0.25 f(1.5, 2.1095) = 0.6016$

$y_2 = 1.5289 + 1/6 (0.5558 + 2(0.5785) + 2(0.5806) + 0.6016)$

$y_2 = 2.1082$

Scanned with CamScanner

e) $y' = xe^{y+x} - 1$, $y(0) = -1$, $0 \le x \le 1$, $h = 0.5$

At $x_0 = 0$, $y_0 = -1$

$\rightarrow$ At $x_1 = 0.5$

$k_0 = 0.5\, f(0, -1) = -0.5$
$k_1 = 0.5\, f(0.25, -1.25) = -0.454$
$k_2 = 0.5\, f(0.25, -1.227) = -0.453$
$k_3 = 0.5\, f(0.5, -1.453) = -0.4036$

$y_1 = -1 + \frac{1}{6}[-0.5 + 2(-0.454) + 2(-0.453) + (-0.4036)]$
$y_1 = \underline{-1.4529}$

$\rightarrow$ At $x_2 = 1$

$k_0 = 0.5\, f(0.5, -1.4529) = -0.4036$
$k_1 = 0.5\, f(0.75, -1.6547) = -0.3488$
$k_2 = 0.5\, f(0.75, -1.6271) = -0.344$
$k_3 = 0.5\, f(1, -1.7969) = -0.2746$

$y_2 = -1.4529 + \frac{1}{6}[-0.4036 + 2(-0.3483) + 2(-0.344) - 0.2746]$
$y_2 = \underline{-1.7967}$

Scanned with CamScanner

Q7 Give answer to the following in one two sentences.

a) What is the Advantages of RK methods over Taylor series methods of the same order?

→ RK methods & Taylor series method of the same order are equality efficient but RK methods having advantage of no requirement of deriving the expressions of deriving the expressions of derivatives

b) What is the order of local truncation error in RK methods of order 2 & order 4? What is the order of global truncation error in these methods?

→ In RK method of order 2 the order of local truncation error is $O(h^3)$. & order of global truncation error is $O(h^2)$.

→ In RK method of order 4 the order of local truncation error is $O(h^5)$, & order of global truncation error is $O(h^4)$.

c) Explain the effect of reducing stepsize in arriving at approximate solution of differential equation at a point

→ Reduction in stepsize yields better results, increases accuracy

Scanned with CamScanner

d) What is the basic difference in method of deriving formulas for Single step methods & multistep methods?

→ Taylor's method is used to derive formula of Singlestep methods while Derivation of multistep methods is based on numerical integration of interpolating polynomial fitted at previous points.

e) What is the main drawback of multistep methods?

→ multistep methods are not self starting.

f) What are the advantages of predictor Corrector methods over RK methods?

→ In predictor Corrector methods only 1/2 is function evaluations per step, depending of number of times, Corrector formulas is applied.

Scanned with CamScanner

Q.8 Consider the IVP
$y' = y - t^2 + 1$, $0 \le t \le 2$, $y(0) = 0.5$
taking $h = 0.2$, exact saluation is
$y(t) = (t+1)^2 - 0.5e^t$

a) Use milne Simpson predictor corrector formula to estimate values at $y(0.8)$ & $y(1.0)$.

→ milene simpson predictor formula is

$y_{i+1} = y_{i-3} + \frac{4h}{3}(2f_i - f_{i-1} + 2f_{i-2})$

→ Correctar formula is

$y_{i+1} = y_{i-1} + \frac{h}{3}(f_{i+1} + 4f_i + f_{i-1})$

now $y' = y - t^2 + 1$, $0 \le t \le 2$, $y(0) = 0.5$, $h = 0.2$
$\therefore t_0 = 0$, $t_1 = 0.2$, $t_2 = 0.4$, $t_3 = 0.6$, $t_4 = 0.8$, $t_5 = 1$

– first we have to estimate values at

ie. at $t_4 = 0.8$

exact sal: is $y(t) = (t+1)^2 - 0.5e^t$
so by exact sal.

$t_0 = 0$, $y_0 = 0.5$
$t_1 = 0.2$, $y_1 = 0.8293$
$t_2 = 0.4$, $y_2 = 1.2141$
$t_3 = 0.6$, $y_3 = 1.6489$
$t_4 = 0.8$, $y_4 = 2.1272$

Scanned with CamScanner

Now by milne simpson predictor formula

$y_4 = y_0 + 4 \times 0.2/3 \, [2f_3 - f_2 + 2f_1]$

$= 0.5 + 0.2667 (2f(t_3, y_3) - (t_2, y_2) + 2(t_1, y_1))$

$= 0.5 + 0.2667 (2((1.6489) - (0.6)^2 + 1) - ((1.2141) - (0.4)^2 + 1) + 2((0.8293) - (0.2)^2 + 1))$

$y_4 = 0.5 + 0.2667 (2 \times 2.289 - 2.0541 + 2 \times 1.7893)$

$y_4 = 2.1275$

→ Now by Corrector formula:

$y_4 = y_2 + 0.2/3 \, (f_4 + 4f_3 + f_2)$

$= 1.2141 + 0.0667 (f(0.8, 2.1275) + 4f(0.6, 1.6489) + f(0.4, 1.2141))$

$y_4 = 2.1277.$

again using Corrector formula:

$y_4 = 1.2141 + 0.0667 (f(0.8, 2.1277) + 4f(0.6, 1.6489) + f(0.4, 1.2141))$

$\underline{y_4 = 2.1277}$

Now to estimate values at $y(1.0)$
i.e. at $t_5 = 1.0$.

By milne simpson predictor formula

$y_5 = y_1 + 4\times0.2/3\ (2f_4 - f_3 + 2f_2)$

$= 0.8293 + 0.2667\ [2\times f\,(0.8, 2.1272) - f\,(0.6, 1.6489)$
$+ 2 f\,(0.4, 1.2141))$

$y_5 = 2.5878$

Now using Corrector formula.

$y_5 = y_3 + 0.2/3\ (f_5 + 4f_4 + f_3)$

$= 1.6489 + 0.0667\ (f(1, 2.5878) + 4f\,(0.8, 2.1272)$
$+ f\,(0.6, 1.6489))$

$y_5 = \underline{2.6378}$

Again,

$y_5 = 1.6489 + 0.0667\ (f\,(1, 2.6378) + 4f\,(0.8, 2.1272)$
$+ f\,(0.6, 1.6489))$

$y_5 = 2.6411$

Again,

$y_5 = \underline{2.6413}$

Scanned with CamScanner

b) Use Adams moulton for Steps predictor corrector method to arrive at approximate solution at $y(0.8)$ & $y(1.0)$.

→ Adams moulton predictor formulas is,

$y_{i+1} = y_i + h/24\,(55f_i - 59f_{i-1} + 37f_{i-2} - 9f_{i-3})$

→ Corrector formula,

$y_{i+1} = y_i + h/24\,(9f(x_{i+1}, y_{i+1}) + 19f_i - 5f_{i-1} + f_{i-2})$

→ we have $y' = y - (t)^2 + 1,\ 0 \le t \le 1,\ y(0) = 0.5,$
$h = 0.2.$

→ Now By exact Sol.

$t_0 = 0,\quad y_0 = 0.5$
$t_1 = 0.2,\quad y_1 = 0.8293$
$t_2 = 0.4,\quad y_2 = 1.2141$
$t_3 = 0.6,\quad y_3 = 1.6489$
$t_4 = 0.8,\quad y_4 = 2.1272$

→ we have to find $y(0.8)$
i.e at $t_4 = 0.8$

$y_4 = y_3 + 0.2/24\,(55f_3 - 59f_2 + 37f_1 - 9f_0)$

$y_4 = 2.1273.$

→ Now By Corrector formula:
$y_4 = y_3 + 0.2/24\,(9f(0.8, 2.1273) + 19f_3 - 5f_2 + f_1)$
$y_4 = \underline{2.1272}$

Scanned with CamScanner

Now to find $y(1.0)$ ie at $t_5 = 1.0$.
By predictor method

$$y_5 = y_4 + \frac{0.2}{24}(55f_4 - 59f_3 + 37f_2 - 9f_1)$$
$$y_5 = 2.6409.$$

By corrector method.

$$y_5 = y_4 + \frac{0.2}{24}(9f(1.0, 2.6409) + 19f_4 - 5f_3 + f_2)$$
$$y_5 = 2.64082$$

(c) Compare the result obtained in (a), (b) from the exact sol by calculating error

	Milne-Simpson	Exact sol.	Error
$y(0.8)$	2.1277	2.1272	0.0005
$y(1.0)$	2.6413	2.6409	0.0323

	Adams-moulton	Exact sol.	Error
$y(0.8)$	2.1272	2.1272	0.0
$y(1.0)$	2.6408	2.6409	0.0001

Scanned with CamScanner

Assignment - 5

Q1) Values for $f(x) = xe^x$ are given below

x	1.8	1.9	2.0	2.1	2.2
$f(x)$	10.889365	12.703199	14.778112	17.148957	17.855030

Use all applicable three point formulas & five point formulas to approximate $f'(2.0)$.

→ Three point endpoint formula.

$f'(x_0) = \frac{1}{2h}[-3f(x_0) + 4f(x_0+h) - f(x_0+2h)]$

Here $h = 0.1$

So

$f'(2.0) = \frac{1}{2\times(0.1)}[-3f(2.0) + 4f(2.1) - f(2.2)]$

$= 5[-3\times(14.778122) + 4\times(17.148957) - (17.855030)]$

$f'(2.0) = 22.03216$

→ Three point midpoint formula

$f'(x_0) = \frac{1}{2h}[-f(x_0-h) + f(x_0+h)]$

$f'(2.0) = 5[-f(1.9) + f(2.1)]$

$= 5[-12.703199 + 17.148957]$

$f'(2.0) = 22.22879$

→ Three point endpoint formula:

$f'(x_0) = \frac{1}{2h}[f(x_0-2h) - 4f(x_0-h) + 3f(x_0)]$

$f'(2.0) = 22.054675$

Teacher's Signature :

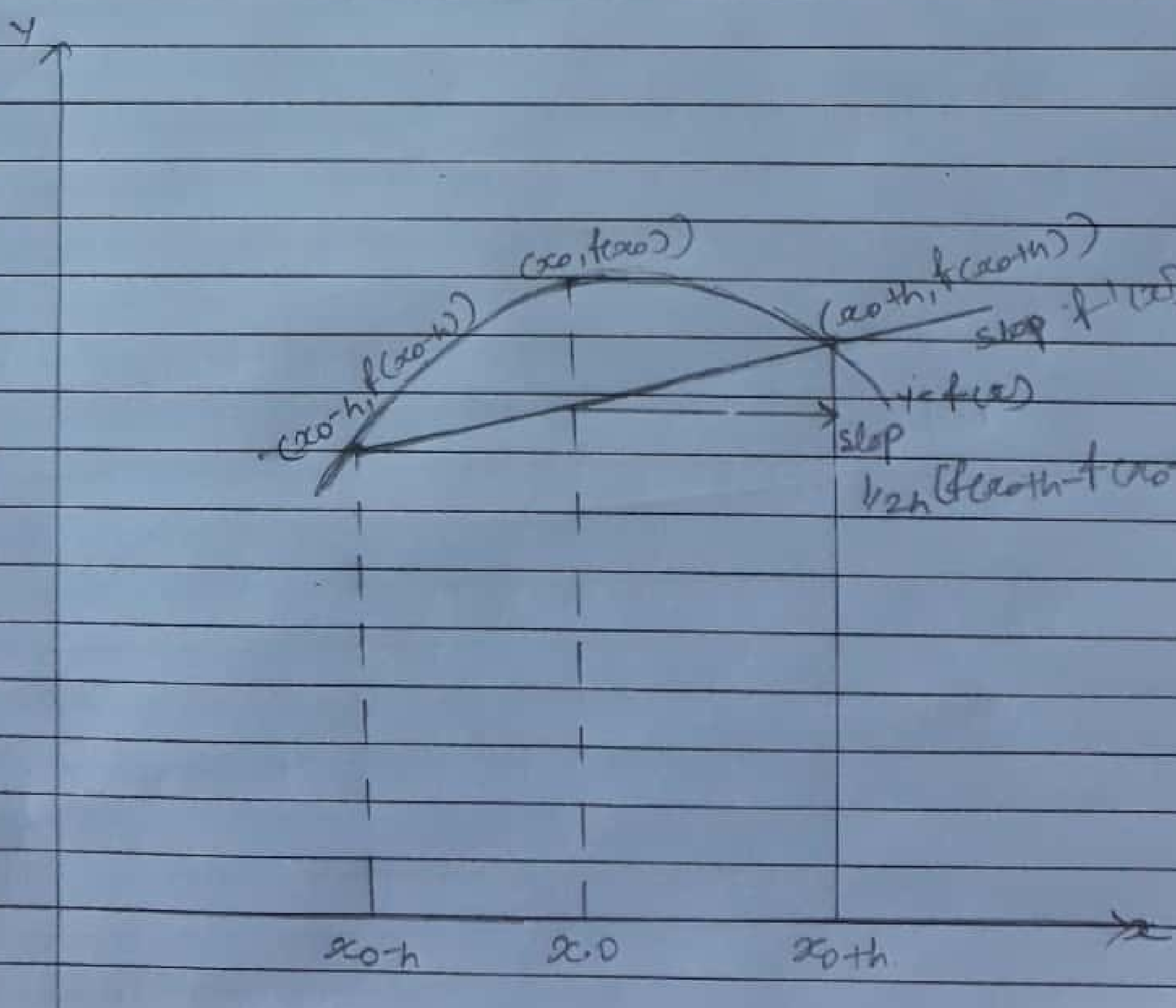
→ five point midpoint formula

$f'(x_0) = \frac{1}{12h}\left[f(x_0-2h) - 8f(x_0-h) + 8f(x_0+h) - f(x_0+2h)\right]$

$f'(2.0) = \frac{1}{12\times 0.1}\left[10.889365 - 8(12.703199) + 8(17.148937) - 19.855030\right]$

$= 22.1669992$

Q.2 Give graphical representation of three point midpoint formula.



Q3 Explain how h too small is not advantages in numerical differentiation?

→ In each of the numerical differentiation formulas, h occurs in denominator. As h is made smaller & smaller to increase accuracy, division by small number causes round off errors to increase. moreover in numerator also, difference of almost equal values occurs, which also contributes to roundoff errors. As a result, beyond a certain value, h can not be reduced further, as round off errors start dominating & accuracy can not be improved further.

Q.4 In a circuit with impressed voltage E(t) and inductance L, kirchhoff's first law gives the relationship.

$$E(t) = L \frac{di}{dt} + Ri,$$

where R is the resistance in the circuit and i is the current. Suppose we measure the current for several values of t and obtain.

t	1.00	1.01	1.02	1.03	1.04
i	3.10	3.12	3.14	3.18	3.24

Where t is measured in seconds, i is in amperes, the inductance L is a constant 0.98 henries & the resistance is 0.142 ohms.
Approximate the voltage $\epsilon(t)$ when $t = 1.00$, 1.01, 1.02, 1.03, 1.04.

→ Here

$\epsilon(t) = L \frac{di}{dt} + Ri$ where $L = 0.98$, $R = 0.142$
$h = 0.01$

To find $\epsilon(t)$ at $t = 1.00$ we need to find di/dt at 1.00.

By three point endpoint formula,

$f'(x_0) = \frac{1}{2h}(-3f(x_0) + 4f(x_0+h) - f(x_0+2h))$

$\therefore f'(1.00) = \frac{1}{2 \times 0.01}[-3(3.10) + 4(3.12) - 3.14)$

$= 50 \times (0.04) = 2.$

So, $\epsilon(1.00) = 0.98 \times 2 + 0.142 \times (3.10)$
$\epsilon(1.00) = \underline{2.4002}$ — ①

→ now at $t = 1.01$
By three point midpoint formula,

$f'(x_0) = \frac{1}{2h}[-f(x_0 - h) + f(x_0+h)]$
$\therefore f(1.01) = 50[-3.10 + 3.14]$
$f(1.01) = \underline{2}$

so,

$E(1.01) = 0.98 \times 2 + 0.142 \times (3.12)$

$E(1.01) = 2.40304$ — (2)

→ now at $t = 1.02$

By five point midpoint formula

→ eqn

$f'(x_0) = \frac{1}{12 \times h}[f(x_0 - 2h) - 8f(x_0 - h) + 8f(x_0 + h) - f(x_0 + 2h)]$

$f'(1.02) = \frac{1}{0.12}[3.10 - 8(3.12) + 8(3.18) - (3.24)]$

$f'(1.02) = 2.83$

so, $E(1.02) = 0.98 \times 2.83 + 0.142 \times (3.14)$

$E(1.02) = 3.21928$ — (3)

→ now at $t = 1.03$

By three point midpoint formula,

$f'(1.03) = \frac{1}{2 \times 0.01}[-3.14 + 3.24]$

$= 5.$

So $E(1.03) = 0.98 \times 5 + 0.142 \times 318$

$E(1.03) = 5.35156$ — (4)

→ now at $t = 1.04$
By three point endpoint formula,

$f'(x_0) = \frac{1}{2h}[f(x_0-2h) - 4f(x_0-h) + 3f(x_0)]$

$f'(1.04) = 50[3.14 - 4(3.18) + 3(3.24)]$
$= 7$

So, $C(1.04) = 0.98 \times 7 + 0.142 \times 3.24$
$C(1.04) = 7.32008$ — ⑤

D E P A R T M E N T O F C O M P U T E R S C I E N C E
R O L L W A L A C O M P U T E R C E N T R E
G U J A R A T U N I V E R S I T Y
M.C.A. - II

R O L L NO : 36
N A M E : Preksha K. Sheth
S U B J E C T : CONM (Computer Oriented Numerical Methods)

NO.	TITLE	PAGE NO.	DATE	SIGN
	Assignment 1			
	1. Apply Bisection method to solve the algebraic equation correct to 6 decimal places.		**01/07/2020**	
	2. Compute the number of iterations needed to find a root correct up to 6 decimal places. Apply Bisection method to solve the equation.		**01/07/2020**	
	3. Apply False Position method to solve the algebraic correct to 6 decimal places.		**01/07/2020**	
	4. Find the root using Secant method.		**01/07/2020**	
	5. Set up a Newton Raphson iteration for computing the square root of a given positive number. Using the same find the square root of 2 exact to six decimal places.		**01/07/2020**	
	6. Find the root using Fixed Point method.		**01/07/2020**	
	Assignment 2			
	1. Evaluate using Trapezoidal rule correct to 3 decimal places.		**01/07/2020**	
	2. Evaluate the integral Using Simpson’s 1/3 Rule and Simpson’s 3/8 rule correct to six decimal places.		**01/07/2020**	
	3. car laps a race track in 84 seconds. The speed of the car at each 6-second interval is determined by using a radar gun and is given from the beginning of the lap, in feet/second by the entries in the following table. How long is the track? Use (i) Trapezoidal Rule (ii) Simpson’s 1/3 rule (iii)Simpson’s 3/8 rule.		**01/07/2020**	
	4. Write a program to solve the differential using Euler’s method and Runge -Kutta second order method and Tabulate your results.		**01/07/2020**	
	5. Find the solution of differential equation using Runge-Kutta method of order 4.		**01/07/2020**	
	6. Find the solution ofdifferential equation using Milne-Simpson’s and Adam-Bashforth-Moulton’s predictor-corrector method.		**01/07/2020**	
	7. Use Adam-Bashforth-Moulton’s predictor-corrector method to obtain the solution of the equation and		**01/07/2020**	

	also using Runge-Kutta second order method. Tabulate the results obtained.			
	8. Use Milne Simpson predictor corrector method to obtain the solution of the equation Compute y(0.1), y(0.2) and y(0.3) using Runge-Kutta fourth order method. Tabulate the results obtained.		**01/07/2020**	
	9. Estimate using 3 point formulas and 5 point formulas From the given table.		**01/07/2020**	

****
****

NAME : Preksha Sheth

ROLL NO : 36

CLASS : MCA-II

SUBJECT : COMPUTER ORIENTED NUMERIC METHODS(CONM)

****
****

ASSIGNMENT - 1

****
****

Q(1): Apply Bisection method to solve the algebraic equation

f(x) = x^3 - x - 1 = 0 correct to 6 decimal places.

****
****

```
#include<stdio.h>

#include<conio.h>

#include<math.h>

#define e 0.0000005

double fun(double a)

{

	double sum;

	//sum=(a*a*a)-(9*a)+1;

	//sum=(3*a)-cos(a)-1;

	//sum=(a*a*a)-9(a)+1;

	sum=(a*a*a)-(a)-1;

	//sum=sin(a)-10*(a-1);

	//sum=a*exp(a)-1;

	return sum;

}

void bisection(double a,double b)

{

	double c;

	int itr=1; //no. itration

	printf("Equation = (a*a*a)-(a)-1 \n");
```

```
	printf("\nEnter First value : ");
	scanf("%lf",&a);
	printf("Enter Second value : ");
	scanf("%lf",&b);
	//c=(a+b)/2;
	while ( (fun(a)*fun(b)) >= 0.0 )
	{
		printf("\nWrong Assumption for Root..Enter Valid Values...");
		printf("\nEnter First value : ");
		scanf("%lf",&a);
		printf("Enter Second value : ");
		scanf("%lf",&b);
	}

	c=(a+b)/2;
	printf("================================================================================
===================");
	printf("\n\nn  \ta  \t\tf(a)  \t\tb  \t\tf(b)  \t\tc  \t\tf(c)\n");
	printf("================================================================================
===================");
	printf("\n%d | \t%lf | \t%lf | \t%lf | \t%lf | \t%lf | \t%lf\n",itr,a,fun(a),b,fun(b),c,fun(c));

	while(fabs(fun(c)) > e && itr < 25)
	{
		//c=(a+b)/2;
		//printf("ds");
		itr++;
		if((fun(c)*fun(a))>=0)
		{
			a=c;
		}
		else
		{
			b=c;
		}
```

```
                //itr++;

                c=(a+b)/2;

                printf("\n%d | \t%lf | \t%lf | \t%lf | \t%lf | \t%lf | \t%lf\n",itr,a,fun(a),b,fun(b),c,fun(c));


                if(fabs(c-b) < e || fabs(c-a) < e)

                {

                        break;

                }

        }

        printf("\n=================================================================================
==============\n");

                printf("Root is %lf ",c);

                //printf("Number of Itration is %d...",itr);

        //printf("%.6lf",b);

}


void main()

{

        double a=0,b=0;

        printf("\n=================================================================================
====================\n");

        printf("\t\t\t\t\t***** Bisection Method *****");

        printf("\n=================================================================================
====================\n");


        bisection(a,b);

}
```

****
****

output:

```
==========================================================================================
                    ***** Bisection Method *****
==========================================================================================
Equation = (a*a*a)-(a)-1


Enter First value : 1

Enter Second value : 2
==========================================================================================
n         a          f(a)            b          f(b)             c          f(c)
==========================================================================================
1 |    1.000000 |    -1.000000 |    2.000000 |    5.000000 |    1.500000 |    0.875000

2 |    1.000000 |    -1.000000 |    1.500000 |    0.875000 |    1.250000 |    -0.296875

3 |    1.250000 |    -0.296875 |    1.500000 |    0.875000 |    1.375000 |    0.224609

4 |    1.250000 |    -0.296875 |    1.375000 |    0.224609 |    1.312500 |    -0.051514

5 |    1.312500 |    -0.051514 |    1.375000 |    0.224609 |    1.343750 |    0.082611

6 |    1.312500 |    -0.051514 |    1.343750 |    0.082611 |    1.328125 |    0.014576

7 |    1.312500 |    -0.051514 |    1.328125 |    0.014576 |    1.320313 |    -0.018711

8 |    1.320313 |    -0.018711 |    1.328125 |    0.014576 |    1.324219 |    -0.002128

9 |    1.324219 |    -0.002128 |    1.328125 |    0.014576 |    1.326172 |    0.006209

10 |    1.324219 |    -0.002128 |    1.326172 |    0.006209 |    1.325195 |    0.002037

11 |    1.324219 |    -0.002128 |    1.325195 |    0.002037 |    1.324707 |    -0.000047

12 |    1.324707 |    -0.000047 |    1.325195 |    0.002037 |    1.324951 |    0.000995
```

```
13 |   1.324707 |    -0.000047 |   1.324951 |    0.000995 |    1.324829 |    0.000474

14 |   1.324707 |    -0.000047 |   1.324829 |    0.000474 |    1.324768 |    0.000214

15 |   1.324707 |    -0.000047 |   1.324768 |    0.000214 |    1.324738 |    0.000084

16 |   1.324707 |    -0.000047 |   1.324738 |    0.000084 |    1.324722 |    0.000018

17 |   1.324707 |    -0.000047 |   1.324722 |    0.000018 |    1.324715 |    -0.000014

18 |   1.324715 |    -0.000014 |   1.324722 |    0.000018 |    1.324718 |    0.000002

19 |   1.324715 |    -0.000014 |   1.324718 |    0.000002 |    1.324717 |    -0.000006

20 |   1.324717 |    -0.000006 |   1.324718 |    0.000002 |    1.324718 |    -0.000002

21 |   1.324718 |    -0.000002 |   1.324718 |    0.000002 |    1.324718 |    0.000000

==========================================================================================
Root is 1.324718 Press any key to continue . . .
```

****

****

Q(2): Compute the number of iterations needed to find a root of f(x) = x*e^x = 1 in the interval (0,1) correct upto 6 decimal places. Apply Bisection method to solve the equation.

****

****

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
#define e 0.0000005
double fun(double a)
{
        double sum;
```

```
	//sum=(a*a*a)-(9*a)+1;
	//sum=(3*a)-cos(a)-1;
	//sum=(a*a*a)-9(a)+1;
	//sum=(a*a*a)-(a)-1;
	//sum=sin(a)-10*(a-1);
	sum=a*exp(a)-1;
	return sum;
}
void bisection(double a,double b)
{
	double c;
	int itr=1,limit; //no. itration
	a=0;
	b=1;
	printf("a = 0\n b = 1\n");
	printf("Equation is a*exp(a)-1\n");
	/*c=(a+b)/2;
	while ((fun(a)*fun(b))>=0 &&)
	{
		printf("\nWrong Assumption for Root..Enter Valid Values...");
		printf("\nEnter First value : ");
		scanf("%lf",&a);
		printf("Enter Second value : ");
		scanf("%lf",&b);
	}*/
	c=(a+b)/2;
	limit= (log(b-a)-log(e)) / (log(2.0));
	printf("Number of Itration = %d\n",limit);
	printf("===========================================================================
===================");
	printf("\n\nn  \ta  \t\tf(a)  \t\tb  \t\tf(b)  \t\tc  \t\tf(c)\n");
	printf("===========================================================================
===================");
	printf("\n%d | \t%.6lf | \t%.6lf | \t%.6lf | \t%.6lf | \t%.6lf | \t%.6lf\n",itr,a,fun(a),b,fun(b),c,fun(c));
```

```
	while(fabs(fun(c)) > e && itr < limit)
	{
		//itr++;
		//c=(a+b)/2;
		//printf("ds");
		//printf("\n%d | \t%.6lf | \t%.6lf | \t%.6lf | \t%.6lf | \t%.6lf |
\t%.6lf\n",itr,a,fun(a),b,fun(b),c,fun(c));
		if( (fun(c)*fun(a)) >= 0)
		{
			a=c;
		}
		else
		{
			b=c;
		}
		itr++;
		c=(a+b)/2;
		printf("\n%d | \t%.6lf | \t%.6lf | \t%.6lf | \t%.6lf | \t%.6lf | \t%.6lf\n",itr,a,fun(a),b,fun(b),c,fun(c));

		if(fabs(c-b) < e || fabs(c-a) < e)
		{
			break;
		}

	}

	printf("\n==================================================================================
==============\n");
		printf("Root is %lf After %d itrations.. ",c,itr);
	//printf("%.6lf",b);
}

void main()
{
	double a=0,b=0;
```

```
	printf("\n===========================================================================
=====================\n");
	printf("\t\t\t\t\t***** Bisection Method *****");
	printf("\n===========================================================================
=====================\n");


	bisection(a,b);
}
```

****
****
output:

```
=============================================================================================
                    ***** Bisection Method *****
=============================================================================================
a = 0
b = 1
Equation is a*exp(a)-1
Number of Itration = 20
=============================================================================================
n         a          f(a)                b          f(b)                c          f(c)
=============================================================================================
1 |    0.000000 |    -1.000000 |    1.000000 |    1.718282 |    0.500000 |    -0.175639

2 |    0.500000 |    -0.175639 |    1.000000 |    1.718282 |    0.750000 |    0.587750

3 |    0.500000 |    -0.175639 |    0.750000 |    0.587750 |    0.625000 |    0.167654

4 |    0.500000 |    -0.175639 |    0.625000 |    0.167654 |    0.562500 |    -0.012782

5 |    0.562500 |    -0.012782 |    0.625000 |    0.167654 |    0.593750 |    0.075142

6 |    0.562500 |    -0.012782 |    0.593750 |    0.075142 |    0.578125 |    0.030619
```

7 | 0.562500 | -0.012782 | 0.578125 | 0.030619 | 0.570313 | 0.008780

8 | 0.562500 | -0.012782 | 0.570313 | 0.008780 | 0.566406 | -0.002035

9 | 0.566406 | -0.002035 | 0.570313 | 0.008780 | 0.568359 | 0.003364

10 | 0.566406 | -0.002035 | 0.568359 | 0.003364 | 0.567383 | 0.000662

11 | 0.566406 | -0.002035 | 0.567383 | 0.000662 | 0.566895 | -0.000687

12 | 0.566895 | -0.000687 | 0.567383 | 0.000662 | 0.567139 | -0.000013

13 | 0.567139 | -0.000013 | 0.567383 | 0.000662 | 0.567261 | 0.000325

14 | 0.567139 | -0.000013 | 0.567261 | 0.000325 | 0.567200 | 0.000156

15 | 0.567139 | -0.000013 | 0.567200 | 0.000156 | 0.567169 | 0.000072

16 | 0.567139 | -0.000013 | 0.567169 | 0.000072 | 0.567154 | 0.000029

17 | 0.567139 | -0.000013 | 0.567154 | 0.000029 | 0.567146 | 0.000008

18 | 0.567139 | -0.000013 | 0.567146 | 0.000008 | 0.567142 | -0.000002

19 | 0.567142 | -0.000002 | 0.567146 | 0.000008 | 0.567144 | 0.000003

20 | 0.567142 | -0.000002 | 0.567144 | 0.000003 | 0.567143 | 0.000000

==

Root is 0.567143 After 20 itrations.. Press any key to continue . . .

****
****

****
****

Q(3): Apply False Position method to solve the algebraic equation

f(x) = x *log10 (x) = 1.2 correct to 6 decimal places.

****
****

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
#define e 0.0000005
double fun(double a)
{
	double sum;
	//sum=(a*a*a)-(9*a)+1;
	//sum=(3*a)-cos(a)-1;
	//sum=(a*a*a)-9(a)+1;
	//sum=(a*a*a)-(a)-1;
	//sum=sin(a)-10*(a-1);
	//sum=a*exp(a)-1;
	sum=(a * (log10(a)) -1.2 );
	return sum;
}
void false_position(double a,double b)
{
	double c;
	int itr=1; //no. itration
	printf("Equation = a * (log10(a)) - 1.2 \n");
	printf("\nEnter First value : ");
	scanf("%lf",&a);
	printf("Enter Second value : ");
	scanf("%lf",&b);
	//c=(a+b)/2;
	while ((fun(a)*fun(b))>=0 && itr < 25)
	{
		printf("\nWrong Assumption for Root..Enter Valid Values...");
```

```
		printf("\nEnter First value : ");
		scanf("%lf",&a);
		printf("Enter Second value : ");
		scanf("%lf",&b);
	}

	c=(b*(fun(a))-a*(fun(b)))/((fun(a))-(fun(b)));
	printf("=================================================================================
====================");
	printf("\n\nn  \ta  \t\tf(a)  \t\tb  \t\tf(b)  \t\tc  \t\tf(c)\n");
	printf("=================================================================================
====================");
	printf("\n%d | \t%lf | \t%lf | \t%lf | \t%lf | \t%lf | \t%lf\n",itr,a,fun(a),b,fun(b),c,fun(c));

	while(fabs(fun(c)) > e)
	{
		//c=(a+b)/2;
		//printf("ds");
		//printf("\n%d | \t%lf | \t%lf | \t%lf | \t%lf | \t%lf | \t%lf\n",itr,a,fun(a),b,fun(b),c,fun(c));
		if((fun(c)*fun(a))>=0)
		{
			a=c;
		}
		else
		{
			b=c;
		}
		itr++;
		c=(b*(fun(a))-a*(fun(b)))/((fun(a))-(fun(b)));
		//printf("\n%d \t %.6lf \t %.6lf \t %.6lf \t %.6lf \t  %.6lf  \t  %.6lf\n",itr,a,fun(a),b,fun(b),c,fun(c));
		printf("\n%d | \t%lf | \t%lf | \t%lf | \t%lf | \t%lf | \t%lf\n",itr,a,fun(a),b,fun(b),c,fun(c));

		if(((fabs(c-a))/c < e || (fabs(c-b))/c < e))
		{
			break;
```

```
		}


	}

	printf("=====================================================================================================");
		printf("\nRoot is %lf using False Position Method....\n",c);
	//printf("%.6lf",b);
}


void main()
{
	double a=0,b=0;
	printf("\n=====================================================================================================\n");
	printf("\t\t\t\t***** False Position Method *****");
	printf("\n=====================================================================================================\n");
	false_position(a,b);
}
```

****

output:

```
===================================================================================
                       ***** False Position Method *****
===================================================================================
Equation = a * (log10(a)) - 1.2


Enter First value : 2
Enter Second value : 3
===================================================================================
n          a          f(a)                b          f(b)                c          f(c)
===================================================================================
1 |    2.000000 |    -0.597940 |    3.000000 |    0.231364 |    2.721014 |    -0.017091


2 |    2.721014 |    -0.017091 |    3.000000 |    0.231364 |    2.740206 |    -0.000384
```

3 | 2.740206 | -0.000384 | 3.000000 | 0.231364 | 2.740636 | -0.000009

4 | 2.740636 | -0.000009 | 3.000000 | 0.231364 | 2.740646 | -0.000000

==

Root is 2.740646 using False Position Method....

Press any key to continue . . .

****

****

Q(4):   Find the root of f(x) = 3x - cosx - 1 using Secant method.

****

****

```
#include<stdio.h>

#include<conio.h>

#include<math.h>

#define e 0.0000005

double fun(double a)

{

        double sum;

        sum = 3 * a - cos(a) - 1;

        return sum;

}


void secant(double a, double b)

{

        int itr=1;

        double c;

        printf("Equation = 3 * a - cos(a) - 1");

        printf("\nEnter First value :");

        scanf("%lf",&a);

        printf("Enter Second value :");

        scanf("%lf",&b);

        //printf("%lf",a);

        while (fun(a) * fun(b) >= 0.0)

        {
```

```
		printf("\nWrong Assumption for Root..");
		printf("\nEnter First value :");
		scanf("%lf",&a);
		printf("Enter Second value :");
		scanf("%lf",&b);
	}
	//printf("Enter Maximum Itration :");
	//scanf("%lf",&limit);
	c = ((b * fun(a) - a * fun(b)) / (fun(a) - fun(b)));
	printf("\n\n n \ta \t\tf(a) \t\tb \t\tf(b) \t\tc \t\tf(c)\n");
	printf("\n=================================================================================
=================\n\n");
	printf("%d |\t%lf |\t%lf |\t%lf |\t%lf |\t%lf |\t%lf\n",itr,a,fun(a),b,fun(b),c,fun(c));
	while (fabs(fun(c)) > e && itr <= 5)
	{
		itr++;
		a=b;
		b=c;
		c = ((b * fun(a) - a * fun(b)) / (fun(a) - fun(b)));
		printf("\n%d |\t%lf |\t%lf |\t%lf |\t%lf |\t%lf |\t%lf\n",itr,a,fun(a),b,fun(b),c,fun(c));

		if(((fabs(c-b)/c)) < e)
		{
			break;
		}
	}
	printf("\n=================================================================================
=================\n\n");
	printf("Root = %lf",c);
}

void main()
{
	double a=0,b=0;
```

```
	printf("\n==============================================================================
=================\n");
	printf("\t\t\t\t\t*** Secant Method ***");
	printf("\n==============================================================================
=================\n\n");
	secant(a,b);
}
```

****
****

output:

```
==========================================================================================
                         *** Secant Method ***
==========================================================================================
Equation = 3 * a - cos(a) - 1
Enter First value :0
Enter Second value :1

 n          a          f(a)              b             f(b)           c          f(c)
==========================================================================================
1 |   0.000000 |    -2.000000 |    1.000000 |    1.459698 |    0.578085 |    -0.103255

2 |   1.000000 |    1.459698 |    0.578085 |    -0.103255 |    0.605959 |    -0.004081

3 |   0.578085 |    -0.103255 |    0.605959 |    -0.004081 |    0.607106 |    0.000014

4 |   0.605959 |    -0.004081 |    0.607106 |    0.000014 |    0.607102 |    -0.000000

==========================================================================================
Root = 0.607102Press any key to continue . . .
```

****

****

Q(5): Set up a Newton Raphson iteration for computing the square root of a given positive number. Using the same find the square root of 2 exact to six decimal places.

```
*******************************************************************************************
*******************************************************************************************
#include<stdio.h>

#include<conio.h>

#include<math.h>

#define e 0.0000005

double fun(double a,double b)

{

	return (a*a)-b;

}

double deriv_fun(double a)

{

	return 2*a;

}


void newton(double a)

{

	double b,c;

	int itr=1,flag = 0;

	printf("Enter the Number For find its Root :");

	scanf("%lf",&b);

	printf("Enter value for a :");

	scanf("%lf",&a);

	c = a - (fun(a,b) / deriv_fun(a));

	printf("\nn \txn \t\tf(xn) \t\tf'(xn) \t\txn+1 \t\tf(xn+1)");

	printf("\n===============================================================================
==");

	printf("\n\n%d \t%lf \t%lf \t%lf \t%lf \t%lf",itr,a,fun(a,b),deriv_fun(a),c,fun(c,b));

	while(fabs(fun(c,b)) >= e && itr <10)

	{

		itr++;

		a = c;
```

```
		if(fabs(deriv_fun(a)) < 0.00001)
		{
			printf("\nDivide by zero Error encountered!!!");
			flag=1;
			break;
		}
		c = a - (fun(a,b) / deriv_fun(a));
		printf("\n\n%d \t%lf \t%lf \t%lf \t%lf \t%lf",itr,a,fun(a,b),deriv_fun(a),c,fun(c,b));
	}
	printf("\n================================================================================");

	if(flag != 1)
	{
		printf("\n\nRoot = %lf\n\n",c);
	}
}

void main()
{
	double a = 0;
	printf("\n================================================================================\n");
	printf("\t\t\t\t*** Newton Raphson ***");
	printf("\n================================================================================\n\n");
	newton(a);
}
```

****
****

output:

```
================================================================================
                    *** Newton Raphson ***
================================================================================
Enter the Number For find its Root :2

Enter value for a :5

n        xn         f(xn)          f'(xn)       xn+1          f(xn+1)
================================================================================


1    5.000000    23.000000    10.000000    2.700000    5.290000


2    2.700000    5.290000    5.400000    1.720370    0.959674


3    1.720370    0.959674    3.440741    1.441455    0.077794


4    1.441455    0.077794    2.882911    1.414471    0.000728


5    1.414471    0.000728    2.828942    1.414214    0.000000
================================================================================
Root = 1.414214
```

****

****

Q(6): Find the root of f(x) = 3x - cosx - 1 using Fixed Point method.

****

****

```
#include<stdio.h>

#include<conio.h>

#include<math.h>

#define e 0.0000005

double gx(double a)//g(x) f(x) = 3x - cosx -1

{

        double sum;

        sum = (1 + cos(a)) / 3;

        return sum;

}
```

```
void fixed_point(double a)
{
	int itr= 1;
	double g;
	printf("\nf(x) = 3x - cosx - 1\n");
	printf("g(x) = (1 + cos(a)) / 3\n");
	printf("Enter Value :");
	scanf("%d",&a);
	printf("\nn \ta \t\tg(a)");
	printf("\n==========================================");
	while(fabs(a - gx(a)) > e && itr < 20)
	{
		printf("\n%d \t%lf \t%lf",itr,a,gx(a));
		a = gx(a);
		itr++;
	}
	printf("\n==========================================");
	printf("\n\nRoot= %lf",a);
}
void main()
{
	double a = 0.0;
	printf("\n============================================================\n");
	printf("\t\t*** Fixed Point ***");
	printf("\n============================================================\n");
	fixed_point(a);
}
```

****
****

```
output:

==============================================================

              *** Fixed Point ***

==============================================================

f(x) = 3x - cosx - 1

g(x) = (1 + cos(a)) / 3

Enter Value :0

n          a          g(a)

============================================

1     0.000000     0.666667

2     0.666667     0.595296

3     0.595296     0.609328

4     0.609328     0.606678

5     0.606678     0.607182

6     0.607182     0.607086

7     0.607086     0.607105

8     0.607105     0.607101

9     0.607101     0.607102

============================================


Root= 0.607102Press any key to continue . . .
```

****
****

****
****

ASSIGNMENT - 2

****
****

Q 1: Evaluate Integral of (e^x^2)*sin x dx from 0 to 1

using Trapezoidal rule correct to 3 decimal places

****
****

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
#define epsilon 0.0005
void trapezoidal(double,double,int);

double f(double x)
{
	return (exp(x*x)*sin(x));
}
void main()
{
	int N=2;
	double a,b;
	a=0;
	b=1;
	trapezoidal(a,b,N);
	getch();
}
void trapezoidal(double a,double b,int N)
{
	int i,limit=20,k=1;
	double sum=0,old_sum=0,h;
	printf("=====================================Trapezoidal
Rule===================================\n\n");
	printf("\nSr No\t\t|\tN\t\t|\th\t\t\t|\tIntegral\n");
```

```
	printf("________________________________________________________________________________________________
________");
	while(k<=limit)
	{
		sum=0;
		h=(b-a)/N;
		for(i=1;i<N;i++)
		{
			sum+=2*f(a+i*h);
		}
		sum+=(f(a)+f(b));
		sum *=h/2;
		printf("\n%d\t\t|\t%d\t\t|\t%lf\t\t|\t%lf",k,N,h,sum);
		if(fabs(sum-old_sum)<epsilon)
		{
			printf("\n\n-->The Estimate of the Integral is %lf",sum);
			break;
		}
		N*=2;
		k++;
		old_sum=sum;
	}
}
```

****
****

output:

```
===================================Trapezoidal Rule=====================================
Sr No		|	N		|	h			|	Integral
___________________________________________________________________________________________
1		|	2		|	0.500000		|	0.879636
2		|	4		|	0.250000		|	0.804736
3		|	8		|	0.125000		|	0.785295
4		|	16		|	0.062500		|	0.780386
5		|	32		|	0.031250		|	0.779156
6		|	64		|	0.015625		|	0.778848
```

-->The Estimate of the Integral is 0.778848

****
****

Q 2 : Evaluate the integral:

integral of dx/(1+x) from 0 to 1

Using

(i) Simpson's 1/3 Rule correct to six decimal places

(ii) Simpson's 3/8 rule correct to six decimal places

****
****

```
#include<stdio.h>

#include<conio.h>

#include<math.h>

#define epsilon 0.0000005


void simpsons1_3(double,double,int);

void simpsons3_8(double,double,int);

double f(double x)

{

        return (1/(1+x));

}


void main()

{

        int N=2;

        double a,b;

        a=0;

        b=1;

        simpsons1_3(a,b,N);

        simpsons3_8(a,b,N);

        getch();

}
```

```
void simpsons1_3(double a,double b,int N)

{

	printf("====================================Simpsons 1/3 Rule==================================\n\n");

	int i,limit=20,k=1;

	double sum=0,old_sum=0,h;

	printf("\nSr No\t\t|\tN\t\t|\th\t\t\t|\tIntegral\n");

	printf("____________________________________________________________________________________________________________");

	while(k<=limit)

	{

		sum=0;

		h=(b-a)/N;

		for(i=1;i<N;i++)

		{

			if(i%2==0)

				sum+=2*f(a+i*h);

			else

				sum+=4*f(a+i*h);

		}

		sum+=(f(a)+f(b));

		sum *=h/3;

		printf("\n%d\t\t|\t%d\t\t|\t%lf\t\t|\t%lf",k,N,h,sum);

		if(fabs(sum-old_sum)<epsilon)

		{

			printf("\n\n-->The Estimate of the Integral Using simpsons1/3 Rule is %lf",sum);

			break;

		}

		N*=2;

		k++;

		old_sum=sum;

	}

	printf("\n\n");

}
```

```
void simpsons3_8(double a,double b,int N)
{
	printf("======================================Simpsons 3/8 Rule====================================\n\n");
	int i,limit=20,k=1;
	double sum=0,old_sum=0,h;
	printf("\nSr No\t\t|\tN\t\t|\th\t\t\t|\tIntegral\n");
	printf("____________________________________________________________________________________________");
	while(k<=limit)
	{
		sum=0;
		h=(b-a)/N;
		for(i=1;i<N;i++)
		{
			if(i%3==0)
				sum+=2*f(a+i*h);
			else
				sum+=3*f(a+i*h);
		}
		sum+=(f(a)+f(b));
		sum *=3*h/8;
		printf("\n%d\t\t|\t%d\t\t|\t%lf\t\t|\t%lf",k,N,h,sum);
		if(fabs(sum-old_sum)<epsilon)
		{
			printf("\n\n-->The Estimate of the Integral Using simpsons3/8 Rule is %lf",sum);
			break;
		}
		N*=2;
		k++;
		old_sum=sum;
	}
}
```

****
****
output:

===================================Simpsons 1/3 Rule===================================

Sr No	N	h	Integral
1	2	0.500000	0.694444
2	4	0.250000	0.693254
3	8	0.125000	0.693155
4	16	0.062500	0.693148
5	32	0.031250	0.693147

-->The Estimate of the Integral Using simpsons1/3 Rule is 0.693147

===================================Simpsons 3/8 Rule===================================

Sr No	N	h	Integral
1	2	0.500000	0.656250
2	4	0.250000	0.660268
3	8	0.125000	0.684854
4	16	0.062500	0.685215
5	32	0.031250	0.691163
6	64	0.015625	0.691186
7	128	0.007813	0.692657
8	256	0.003906	0.692658
9	512	0.001953	0.693025
10	1024	0.000977	0.693025

-->The Estimate of the Integral Using simpsons3/8 Rule is 0.693025

****
****

Q 3 : A car laps a race track in 84 seconds. The speed of the car at each 6-second interval is determined by using a radar gun and is given from the beginning of the lap, in feet/second by the entries in the following table.

Time 0 6 12 18 24 30 36 42 48 54 60 66 72 78 84

Speed 124 134 148 156 147 133 121 109 99 85 78 89 104 116 123

How long is the track?

Use (i) Trapezoidal Rule (ii) Simpson's 1/3 rule (iii) Simpson's3/8 rule

****
****

```
#include<stdio.h>
#include<conio.h>
#include<math.h>

void simpsons1_3(double,double,int);
void simpsons3_8(double,double,int);
void trapezoidal(double,double,int);

double f(int x)
{
	switch(x)
	{
		case 0:return 124;
		case 6:return 134;
		case 12:return 148;
		case 18:return 156;
		case 24:return 147;
		case 30:return 133;
		case 36:return 121;
		case 42:return 109;
		case 48:return 99;
		case 54:return 85;
		case 60:return 78;
		case 66:return 89;
```

```
			case 72:return 104;
			case 78:return 116;
			case 84:return 123;
		}
}
void main()
{
		int N=14;
		double a,b;
		a=0;
		b=84;
		trapezoidal(a,b,N);
		simpsons1_3(a,b,N);
		simpsons3_8(a,b,N);
		getch();
}

void trapezoidal(double a,double b,int N)
{
		int i;
		double sum=0,h;
		sum=0;
		h=(b-a)/N;
		for(i=1;i<N;i++)
		{
			sum+=2*f(a+i*h);
		}
		sum+=(f(a)+f(b));
		sum *=h/2;
		printf("\n-->Length of track using Trapezoidal Rule=%0.2lf Feet",sum);
}

void simpsons1_3(double a,double b,int N)
{
```

```
    int i;
    double sum=0,h;
        sum=0;
        h=(b-a)/N;
        for(i=1;i<N;i++)
        {
            if(i%2==0)
                sum+=2*f(a+i*h);
            else
                sum+=4*f(a+i*h);
        }
        sum+=(f(a)+f(b));
        sum *=h/3;
        printf("\n-->Length of track using Simpsons 1/3 Rule=%0.2lf Feet",sum);
}

void simpsons3_8(double a,double b,int N)
{
    int i;
    double sum=0,h;
    sum=0;
    h=(b-a)/N;
    for(i=1;i<N;i++)
    {
        if(i%3==0)
            sum+=2*f(a+i*h);
        else
            sum+=3*f(a+i*h);
    }
    sum+=(f(a)+f(b));
    sum *=3*h/8;
    printf("\n-->Length of track using Simpsons 3/8 Rule=%0.2lf Feet",sum);
}
```

****
****

output:

-->Length of track using Trapezoidal Rule=9855.00 Feet

-->Length of track using Simpsons 1/3 Rule=9858.00 Feet

-->Length of track using Simpsons 3/8 Rule=9760.50 Feet

****
****

Q 4 : Write a program to solve the differential equation dy/dx=(y-x)/(y+x), where y(0) = 1, using

(i) Euler's method

(ii) Runge - Kutta second order method

in the interval 0 to 1 using step-size 0.1 Tabulate your results

****
****

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
void euler(double,double,double,int);
void runge_kutta_2(double,double,double,int);

double f(double x,double y)
{
	return ((y-x)/(y+x));
}
void main()
{
	int limit;
	double xi,yi,h;
	xi=0;
	yi=1;
	h=0.1;
	limit=1;
	euler(xi,yi,h,limit);
	runge_kutta_2(xi,yi,h,limit);
```

```
        getch();
}

void euler(double xi,double yi,double h,int limit)
{
        double yi_1;
        yi_1=yi;
        printf("=============EULER METHOD============\n\n");
        printf("\nx\t\t|\tSolution\n");
        printf("____________________________________\n");
        while(xi<=limit)
        {
                yi=yi_1;
                printf("\n%0.2lf\t\t|\t%lf",xi,yi);
                yi_1=yi + h* f(xi,yi);
                xi+=h;
        }
        printf("\n\n-->Solution With Eulers method= %lf\n\n",yi);
}

void runge_kutta_2(double xi,double yi,double h,int limit)
{
        double yi_1,k0,k1;
        yi_1=yi;
        printf("=======RUNGE-KUTTA SECOND ORDER METHOD=======\n\n");
        printf("\nx\t\t|\tSolution\n");
        printf("____________________________________");
        while(xi<=limit)
        {
                yi=yi_1;
                printf("\n%0.2lf\t\t|\t%lf",xi,yi);
                k0=h*f(xi,yi);
                k1=h*f(xi+h,yi+k0);
                yi_1=yi + (0.5)*(k0+k1);
```

```
		xi+=h;
	}
	printf("\n\n-->Solution With RUNGE-KUTTA SECOND ORDER METHOD= %lf",yi);
}
```

****

output:

```
=============EULER METHOD============
x		|	Solution
___________________________________
0.00		|	1.000000
0.10		|	1.100000
0.20		|	1.183333
0.30		|	1.254418
0.40		|	1.315818
0.50		|	1.369193
0.60		|	1.415694
0.70		|	1.456161
0.80		|	1.491231
0.90		|	1.521399
1.00		|	1.547062

-->Solution With Eulers method= 1.547062

=======RUNGE-KUTTA SECOND ORDER METHOD=======

x		|	Solution
___________________________________
0.00		|	1.000000
0.10		|	1.091667
0.20		|	1.168728
0.30		|	1.234629
0.40		|	1.291489
```

0.50 | 1.340729

0.60 | 1.383361

0.70 | 1.420135

0.80 | 1.451627

0.90 | 1.478291

1.00 | 1.500491

-->Solution With RUNGE-KUTTA SECOND ORDER METHOD= 1.500491

****
****

Q 5 : Find the solution of differential equation, for the range 0 <= t <= 1 dy/dt = t + (y)^(1/2) with y(0) = 1, taking step size h = 0.2 using Runge-Kutta method of order 4

****
****

```
#include<stdio.h>
#include<conio.h>
#include<math.h>

void runge_kutta_4(double,double,double,int);

double f(double t,double y)
{
	return (t+sqrt(y));
}
void main()
{
	int limit;
	double ti,yi,h;
	ti=0;
	yi=1;
	h=0.2;
	limit=1;
	runge_kutta_4(ti,yi,h,limit);
	getch();
}
```

```
void runge_kutta_4(double ti,double yi,double h,int limit)
{
	double yi_1,k0,k1,k2,k3;
	yi_1=yi;
	printf("=======RUNGE-KUTTA FORTH ORDER METHOD=======\n\n");
	printf("\nt\t\t|\tSolution\n");
	printf("___________________________________");
	while(ti<=limit)
	{
		yi=yi_1;
		printf("\n%0.2lf\t\t|\t%lf",ti,yi);
		k0=h*f(ti,yi);
		k1=h*f(ti+(h/2),yi+(k0/2));
		k2=h*f(ti+(h/2),yi+(k1/2));
		k3=h*f(ti+h,yi+k2);
		yi_1=yi + (k0+2*k1+2*k2+k3)/6;
		ti+=h;
	}
	printf("\n\n-->Solution With RUNGE-KUTTA FORTH ORDER METHOD= %lf",yi);
}
```

****
****

output:

```
=======RUNGE-KUTTA FORTH ORDER METHOD=======

t		|	Solution
___________________________________
0.00		|	1.000000
0.20		|	1.230632
0.40		|	1.524809
0.60		|	1.885413
0.80		|	2.314716
1.00		|	2.814506

-->Solution With RUNGE-KUTTA FORTH ORDER METHOD= 2.814506
```

****
****

Q 6 : Find the solution of differential equation dy/dt = 1/2 (t+y), for y (2.0) given

y(0) = 2

y(0.5) = 2.636

y(1.0) = 3.595

and y(1.5) = 4.968 , use h = 0.5

using (i) Milne-Simpson’s predictor corrector method

(ii) Adam-Bashforth-Moulton’s predictor-corrector method

****
****

```
#include<stdio.h>

#include<conio.h>

#include<math.h>

#define epsilon 0.00005

void milne_simpsone_predictor_corrector(double[],double[],double);

void adam_bashforth_moultons_predictor_corrector(double[],double[],double);


double f(double t,double y)

{

	return ((t+y)/2);

}

void main()

{

	double h,y[10],t[10];

	h=0.5;

	y[0]=2;

	y[1]=2.636;

	y[2]=3.595;

	y[3]=4.968;

	t[0]=0;

	t[1]=0.5;

	t[2]=1.0;

	t[3]=1.5;
```

```
	t[4]=2.0;
	milne_simpsone_predictor_corrector(y,t,h);
	adam_bashforth_moultons_predictor_corrector(y,t,h);
	getch();
}
void milne_simpsone_predictor_corrector(double y[],double t[],double h)
{
	double yi_old=0;
	int i;
	i=3;
	printf("========milne_simpsone_predictor_corrector METHOD========\n\n");
	//predictor Method
	y[i+1]=y[i-3]+(4*h)*(2*f(t[i],y[i])-f(t[i-1],y[i-1])+2*f(t[i-2],y[i-2]))/3;
	printf("Using Predictor Formula y4 =%lf",y[i+1]);


	//Corrector formula
	while(fabs(yi_old-y[i+1])>epsilon)
	{
		yi_old=y[i+1];
		y[i+1]=y[i-1] + (h/3) *(f(t[i+1],y[i+1])+ 4* f(t[i],y[i])+f(t[i-1],y[i-1]));
		printf("\n-->Using Corrector Formula y4=%lf",y[i+1]);
	}
	printf("\n\n---->Solution With milne_simpsone_predictor_corrector METHOD= %lf\n\n",y[i+1]);
}

void adam_bashforth_moultons_predictor_corrector(double y[],double t[],double h)
{
	double yi_old=0;
	int i;
	i=3;
	printf("========adam_bashforth_moultons_predictor_corrector METHOD========\n\n");
	//predictor Method
	y[i+1]=y[i]+(h/24)*(55*f(t[i],y[i])-59*f(t[i-1],y[i-1])+37*f(t[i-2],y[i-2])-9*f(t[i-3],y[i-3]));
```

```
	printf("Using Predictor Formula y4 =%lf",y[i+1]);



	//Corrector formula
	while(fabs(yi_old-y[i+1])>epsilon)
	{
		yi_old=y[i+1];
		y[i+1]=y[i] + (h/24) *(9*f(t[i+1],y[i+1])+ 19 * f(t[i],y[i])-5*f(t[i-1],y[i-1])+f(t[i-2],y[i-2]));
		printf("\n-->Using Corrector Formula y4=%lf",y[i+1]);
	}
	printf("\n\n---->Solution With adam_bashforth_moultons_predictor_corrector METHOD= %lf",y[i+1]);
}
```

****
****

output:

```
========milne_simpsone_predictor_corrector METHOD========

Using Predictor Formula y4 =6.871000
-->Using Corrector Formula y4=6.873167
-->Using Corrector Formula y4=6.873347
-->Using Corrector Formula y4=6.873362

---->Solution With milne_simpsone_predictor_corrector METHOD= 6.873362

========adam_bashforth_moultons_predictor_corrector METHOD========

Using Predictor Formula y4 =6.870781
-->Using Corrector Formula y4=6.873104
-->Using Corrector Formula y4=6.873322
-->Using Corrector Formula y4=6.873343

---->Solution With adam_bashforth_moultons_predictor_corrector METHOD= 6.873343
```

****
****

Q 7 : Use Adam-Bashforth-Moulton's predictor-corrector method to obtain the solution of the equation dy/dx= 1-xy/x^2 at x = 1.4, where y(1) = 1.

Compute y(1.1), y(1.2) and y(1.3) using Runge-Kutta second order method.

Tabulate the results obtained thus.

****
****

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
#define epsilon 0.00005

void adam_bashforth_moultons_predictor_corrector(double[],double[],double);
double runge_kutta_2(double,double,double,double);
double f(double x,double y)
{
	return ((1-x*y)/(x*x));
}
void main()
{
	double h,y[10],x[10];
	h=0.1;
	x[0]=1;
	x[1]=1.1;
	x[2]=1.2;
	x[3]=1.3;
	x[4]=1.4;

	y[0]=1;
	y[1]=runge_kutta_2(x[0],y[0],h,1.2);
	y[2]=runge_kutta_2(x[0],y[0],h,1.3);
	y[3]=runge_kutta_2(x[0],y[0],h,1.4);

	printf("\n=======================By Runge-Kutta second order method\n");
	printf("y(1.1)=%lf\ny(1.2)=%lf\ny(1.3)=%lf\n\n",y[1],y[2],y[3]);
```

```
	adam_bashforth_moultons_predictor_corrector(y,x,h);
	getch();
}

void adam_bashforth_moultons_predictor_corrector(double y[],double x[],double h)
{
	double yi_old=0;
	int i;
	i=3;
	printf("========adam_bashforth_moultons_predictor_corrector METHOD========\n\n");
	//predictor Method
	y[i+1]=y[i]+(h/24)*(55*f(x[i],y[i])-59*f(x[i-1],y[i-1])+37*f(x[i-2],y[i-2])-9*f(x[i-3],y[i-3]));
	printf("Using Predictor Formula y(1.4) =%lf",y[i+1]);



	//Corrector formula
	while(fabs(yi_old-y[i+1])>epsilon)
	{
		yi_old=y[i+1];
		y[i+1]=y[i] + (h/24) *(9*f(x[i+1],y[i+1])+ 19 * f(x[i],y[i])-5*f(x[i-1],y[i-1])+f(x[i-2],y[i-2]));
		printf("\n-->Using Corrector Formula y(1.4)=%lf",y[i+1]);
	}
	printf("\n\n---->Solution With adam_bashforth_moultons_predictor_corrector METHOD= %lf",y[i+1]);
}

double runge_kutta_2(double xi,double yi,double h,double limit)
{
	double yi_1,k0,k1;
	yi_1=yi;
	while(xi<limit)
	{
		yi=yi_1;
		k0=h*f(xi,yi);
		k1=h*f(xi+h,yi+k0);
```

```
                yi_1=yi + (0.5)*(k0+k1);
                xi+=h;
        }
        return yi;
}
```

****
****

```
output:
========================By Runge-Kutta second order method
y(1.1)=0.995868
y(1.2)=0.985480
y(1.3)=0.971311

========adam_bashforth_moultons_predictor_corrector METHOD========

Using Predictor Formula y(1.4) =0.954695
-->Using Corrector Formula y(1.4)=0.954878
-->Using Corrector Formula y(1.4)=0.954873

---->Solution With adam_bashforth_moultons_predictor_corrector METHOD= 0.954873
```

****
****

Q 8 : Use Milne Simpson predictor corrector method to obtain the solution of
the equation dy/dx= 1-xy/x^2 at x = 1.4, where y(1) = 1.
Compute y(1.1), y(1.2) and y(1.3) using Runge-Kutta fourth order method.
Tabulate the results obtained thus.

****
****

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
#define epsilon 0.00005
```

```
void milne_simpsone_predictor_corrector(double[],double[],double);
double runge_kutta_4(double,double,double,double);
double f(double x,double y)
{
	return ((1-x*y)/(x*x));
}
void main()
{
	double h,y[10],x[10];
	h=0.1;
	x[0]=1;
	x[1]=1.1;
	x[2]=1.2;
	x[3]=1.3;
	x[4]=1.4;

	y[0]=1;
	y[1]=runge_kutta_4(x[0],y[0],h,1.2);
	y[2]=runge_kutta_4(x[0],y[0],h,1.3);
	y[3]=runge_kutta_4(x[0],y[0],h,1.4);

	printf("\n========================By Runge-Kutta Forth order method\n");
	printf("y(1.1)=%lf\ny(1.2)=%lf\ny(1.3)=%lf\n\n",y[1],y[2],y[3]);
	milne_simpsone_predictor_corrector(y,x,h);
	getch();
}
void milne_simpsone_predictor_corrector(double y[],double x[],double h)
{
	double yi_old=0;
	int i;
	i=3;
	printf("========milne_simpsone_predictor_corrector METHOD========\n\n");
	//predictor Method
	y[i+1]=y[i-3]+(4*h)*(2*f(x[i],y[i])-f(x[i-1],y[i-1])+2*f(x[i-2],y[i-2]))/3;
```

```
	printf("Using Predictor Formula y(1.4)=%lf",y[i+1]);



	//Corrector formula
	while(fabs(yi_old-y[i+1])>epsilon)
	{
		yi_old=y[i+1];
		y[i+1]=y[i-1] + (h/3) *(f(x[i+1],y[i+1])+ 4* f(x[i],y[i])+f(x[i-1],y[i-1]));
		printf("\n-->Using Corrector Formula y(1.4)=%lf",y[i+1]);
	}
	printf("\n\n---->Solution With milne_simpsone_predictor_corrector METHOD= %lf\n\n",y[i+1]);
}
double runge_kutta_4(double xi,double yi,double h,double limit)
{
	double yi_1,k0,k1,k2,k3;
	yi_1=yi;
	while(xi<limit)
	{
		yi=yi_1;
		k0=h*f(xi,yi);
		k1=h*f(xi+(h/2),yi+(k0/2));
		k2=h*f(xi+(h/2),yi+(k1/2));
		k3=h*f(xi+h,yi+k2);
		yi_1=yi + (k0+2*k1+2*k2+k3)/6;
		xi+=h;
	}
	return yi;
}
```

****
****

output:

```
========================By Runge-Kutta Forth order method
y(1.1)=0.995737
y(1.2)=0.985268
y(1.3)=0.971050

========milne_simpsone_predictor_corrector METHOD========

Using Predictor Formula y(1.4)=0.954478
-->Using Corrector Formula y(1.4)=0.954629
-->Using Corrector Formula y(1.4)=0.954626

---->Solution With milne_simpsone_predictor_corrector METHOD= 0.954626
```

****
****

Q 9 : From the following table estimate y'(1.1) and y'(1.2) using 3 point formulas and 5 point formulas

x 1.0 1.05 1.10 1.15 1.20 1.25 1.30

y 1.0 1.0247 1.0488 1.0724 1.0954 1.1180 1.1402

****
****

```
#include<stdio.h>
#include<conio.h>
#include<math.h>

void _3point_formulas(double[],double[],double);
void _5point_formulas(double[],double[],double);

void main()
{
        double x[10],y[10],h=0.5;
        x[0]=1.0;
        x[1]=1.05;
        x[2]=1.10;
```

```
    x[3]=1.15;
    x[4]=1.20;
    x[5]=1.25;
    x[6]=1.30;
    y[0]=1.0;
    y[1]=1.0247;
    y[2]=1.0488;
    y[3]=1.0724;
    y[4]=1.0954;
    y[5]=1.1180;
    y[6]=1.1402;

    _3point_formulas(x,y,h);
    _5point_formulas(x,y,h);
    getch();
}

void _3point_formulas(double x[],double y[],double h)
{

    double x0=x[2],ans;
    int i=2;
    //Endpoint formula
    printf("\n===============3 Pont End Point Formula===============\n");
    ans=(1/(2*h)) * (-3 * y[i] + 4*y[i+1]-y[i+2]);
    printf("\n--->y(1.1)'=%lf",ans);
    i=4;
    ans=(1/(2*h)) * (-3 * y[i] + 4*y[i+1]-y[i+2]);
    printf("\n--->y(1.2)'=%lf",ans);
    //Midpoint Formula
    i=2;
    printf("\n===============3 Pont Mid Point Formula===============\n");
    ans=(1/(2*h)) * (-y[i-1] + y[i+1]);
    printf("\n--->y(1.1)'=%lf",ans);
```

```
	i=4;
	ans=(1/(2*h)) * (-y[i-1] + y[i+1]);
	printf("\n--->y(1.2)'=%lf",ans);
	//Endpoint formula
	printf("\n===============3 Pont End Point Formula===============\n");
	i=2;
	ans=(1/(2*h)) * (y[i-2] - 4*y[i-1]+3*y[i]);
	printf("\n--->y(1.1)'=%lf",ans);
	i=4;
	ans=(1/(2*h)) * (y[i-2] - 4*y[i-1]+3*y[i]);
	printf("\n--->y(1.2)'=%lf",ans);
}


void _5point_formulas(double x[],double y[],double h)
{

	double x0=x[2],ans;
	int i=2;
	//Endpoint formula
	printf("\n\n\n===============5 Pont End Point Formula===============\n");
	ans=(1/(12*h)) * ( -25*y[i] +48*y[i+1]-36* y[i+2]+16*y[i+3]-3*y[i+4]);
	printf("\n--->y(1.1)'=%lf",ans);

	//Midpoint Formula
	i=2;
	printf("\n===============5 Pont Mid Point Formula===============\n");
	ans=(1/(12*h)) * ( y[i-2] - 8*y[i-1]+8* y[i+1]-y[i+2]);
	printf("\n--->y(1.1)'=%lf",ans);
	i=4;
	ans=(1/(12*h)) * ( y[i-2] - 8*y[i-1]+8* y[i+1]-y[i+2]);
	printf("\n--->y(1.2)'=%lf",ans);
}
```

****
****

output:

===============3 Pont End Point Formula===============

--->y(1.1)'=0.047800

--->y(1.2)'=0.045600

===============3 Pont Mid Point Formula===============

--->y(1.1)'=0.047700

--->y(1.2)'=0.045600

===============3 Pont End Point Formula===============

--->y(1.1)'=0.047600

--->y(1.2)'=0.045400

===============5 Pont End Point Formula===============

--->y(1.1)'=0.048033

===============5 Pont Mid Point Formula===============

--->y(1.1)'=0.047700

--->y(1.2)'=0.045567

****
****