Attempt all the questions:

Q.N.1) Define the fixed point iteration method. Given the function f(x) = x^2-2x-3=0, rearrange the function in such a way that the iteration method converses to its roots.
(2+3+3)

To find the root of $f(x) = 0$, we first re arrange this into an equivalent form $x - g(x) = 0$, or $x = g(x)$. So, c is a root of $f(x)$, i.e. $f(c) = 0$ if and only if $c = g(c)$. Such a point c is called the fixed point or root of the equation $x = g(x)$. Therefore finding the root of $f(x) = 0$ is equivalent to finding the fixed point of $x = g(x)$. The fixed point of $x = g(x)$ is found iteratively as follows:

An initial guess x_0 is made which is then used to get the next approximation as $x_1 = g(x_0)$. This is then used to obtain $x_2 = g(x_1)$. This iterative process can be expressed in general form as:

$$x_{n+1} = g(x_n)\ ; n = 0, 1, 2, \ldots$$

We continue this process until the fixed point or a sufficient approximation of it is found.

<u>Numerical:</u>

Consider the equation $f(x) = x^2 - 2x - 3 = 0$ whose roots are $x = 1$ and $x = 3$. Then the above equation can be written as

$$x = g_1(x) = \sqrt{2x + 3} \quad \ldots\ldots\ldots (i)$$

$$x = g_2(x) = \frac{x^2 - 3}{2} \quad \ldots\ldots\ldots (ii)$$

$$x = g_3(x) = \frac{3}{x - 2} \quad \ldots\ldots\ldots (iii)$$

If we start with $x_0 = 4$ and the iteration $x_{n+1} = \sqrt{2x_n + 3}$ obtained from (i), we get the following values.

$x_0 = 4$

$x_1 = \sqrt{2 \times 4 + 3} = \sqrt{11} = 3.31662$

$x_2 = \sqrt{2 \times 3.31662 + 3} = 3.10375$

$x_3 = 3.03439$

$x_4 = 3.01144$

$x_5 = 3.00381$

Therefore, $x_n's$ converge to the root $x = 3$ in this case.

If we start with $x_0 = 4$ and the iteration with $x_{n-1} = \frac{3}{x_n - 2}$ obtained from (iii) we get the values

$x_0 = 4$

$$x_1 = \frac{3}{4-2} = 1.5$$

$x_2 = -6$

$x_3 = -0.375$

$x_4 = -1.263158$

$x_5 = -0.919355$

$x_6 = -1.02762$

$x_7 = -0.990876$

$x_8 = -1.00305$

Therefore, the $x_n's$ converge to the root $x = -1$

Q.N.2) What do you mean by interpolation problem? Define divided difference table & construct the table from the following data set. (2+2+4)

X	3.2	2.7	1.0	4.8	5.6
F	22.0	17.8	14.2	38.3	51.7

Suppose we are given a set of $n+1$ data points $(x_i, f_i), i = 0, 1, 2, \ldots, n$. Then interpolation is the method of finding a function $f(x)$ called an interpolation function, such that $f(x) = f_i, 0 \leq i \leq n$ and estimating the value of f by $f(x)$for some x lying in between $x_0, x_1, x_2, \ldots, x_n$.

Divided difference table is a recursive division process whose k^{th} degree polynomial approximation to $f(x)$ can be written as

$$f(x) = f[x_0] + (x - x_0)f[x_0, x_1] + (x - x_0)(x - x_1)f[x_0, x_1, x_2] + \cdots \\ + (x - x_0)(x - x_1) \ldots (x - x_{k-1})f[x_0, x_1, \ldots, x_k]$$

The divided difference table from the given data sets can be constructed as follows:

x_i	3.2	2.7	1.0	4.8	5.6
$f[x_i]$	22.0	17.8	14.2	38.3	51.7
$f[x_i, x_{i+1}]$	8.4	2.12	6.34	16.75	

$f[x_i, x_{i+1}, x_{i+2}]$	2.85 2.01 2.26
$f[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$	-0.52 0.09
$f[x_i, x_{i+1}, x_{i+2}, x_{i+3}, x_{i+4}]$	0.25

OR

Find the least squares line that fits the following data.

X	1	2	3	4	5	6
Y	**5.04**	**8.12**	**10.64**	**13.18**	**16.20**	**20.04**

What do you mean by least squares approximation?

Here, $n = 6$

x_i	y_i	x_i^2	$x_i y_i$
1	5.04	1	5.04
2	8.12	4	16.24
3	10.64	9	31.92
4	13.18	16	52.72
5	16.2	25	81
6	20.04	36	120.24
$\sum x_i = 21$	$\sum y_i = 73.22$	$\sum x_i^2 = 91$	$\sum x_i y_i = 307.16$

Now,

$$b = \frac{6 \times 307.16 - 21 \times 73.22}{6 \times 91 - 21^2} = \frac{305.34}{105} = 2.91$$

And

$$a = \frac{73.22 - 2.91 \times 21}{6} = \frac{12.11}{6} = 2.02$$

Therefore, the least square line that can be fit is $y = 2.02 + 2.91x$

Least Square Approximation (LSA):

Let $(x_i, y_i), i = 1, 2, 3, \ldots, n$ be a given set of n pairs of data points and let $y = f(x)$ be the curve that is fitted to this data. At $x = x_i$, the given value of the ordinate is y_i and the corresponding value on the fitting curve is $f(x_i)$. Then the error of approximating at $x = x_i$ is

$$e_i = y_i - f(x_i)$$

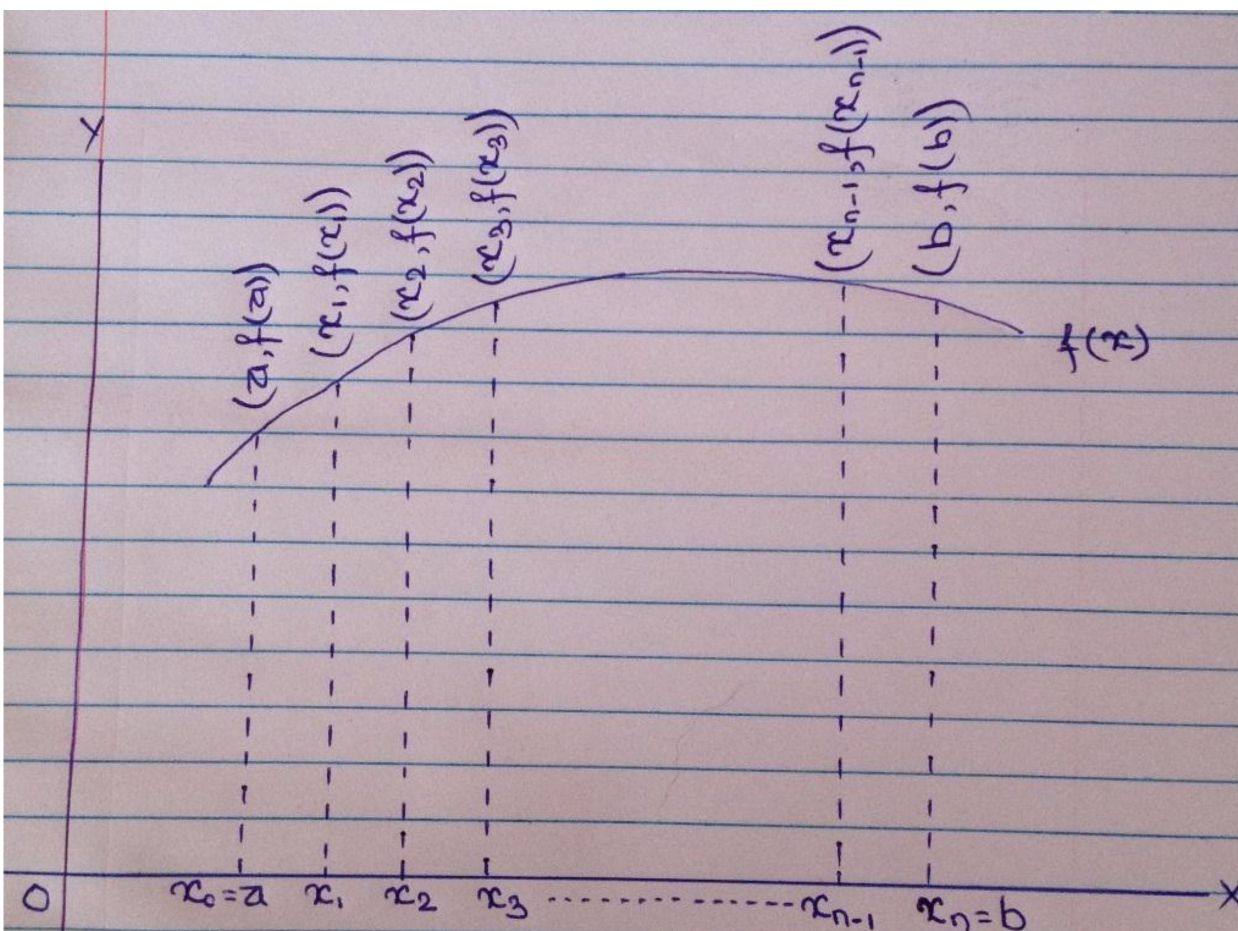
Let

$$S = \left(y_1 - f(x_1)\right)^2 + \left(y_2 - f(x_2)\right)^2 + \cdots + \left(y_n - f(x_n)\right)^2 = e_1^2 + e_2^2 + \cdots + e_n^2 = \sum_{i=1}^{n} e_i^2$$

Then the method of least squares approximation consists of finding an expression $y = f(x)$ such that the sum of the squares of the errors is minimized. i.e. S is minimum.

Q.N.3) Derive a composite formula of the trapezoidal rule with its geometrical figure. Evaluate $\int_0^1 e^{-x2} dx$ using this rule with n=5, up to 6 decimal places. (4+4)

Composite trapezoidal rule:



Suppose we have to evaluate the integral $\int_a^b f(x)dx$. We first divide the interval $[a, b]$ into n equal spaced sub-intervals by points $x_1 = x + ih$, where $i = 0, 1, 2, \ldots, n$ & $h = \frac{b-a}{n}$. Then in each sub-interval $[x_{i-1}, x_i], i = 1, 2, 3, \ldots, n$. We approximate the integral $\int_{x_{i-1}}^{x_i} f(x)dx$ by the trapezoidal formula $\frac{h}{2}[f(x_{i-1}) + f(x_i)]$ so that

$$\int_a^b f(x)dx = \int_{x_0}^{x_1} f(x)dx + \int_{x_1}^{x_2} f(x)dx + \cdots + \int_{x_{n\_1}}^{x_n} f(x)dx$$

$$= \frac{h}{2}[f(x_0) + f(x_1)] + \frac{h}{2}[f(x_1) + f(x_2)] + \cdots + \frac{h}{2}[f(x_{n-1}) + f(x_n)]$$

$$= \frac{h}{2}\left[f(x_0) + 2\big(f(x_1) + f(x_2) + \cdots + f(x_{n-1})\big) + f(x_n)\right]$$

Therefore,

$$\int_a^b f(x)dx = \frac{h}{2}[f_0 + 2(f_1 + f_2 + \cdots + f_{n-1}) + f_n]$$

Which is the composite trapezoidal rule for calculating $\int_a^b f(x)dx$.

<u>Numerical:</u>

Here, $a = 0\,; b = 1\,; n = 5$

So, $h = \frac{b-a}{n} = 0.2$

Now we get the following table

x	0	0.2	0.4	0.6	0.8	1.0
$f(x)$	1	0.960789	0.852144	0.697676	0.527292	0.367879

Therefore, the value of $\int_0^1 e^{-x^2}\,dx$ using composite trapezoidal rule is

$$\int_0^1 e^{-x^2}\,dx = \frac{0.2}{2}[1 + 2(0.960789 + 0.852144 + 0.697676 + 0.527292) + 0.367879]$$
$$= 0.7443681$$

Q.N.4) Solve the following system of algebraic linear equation using Jacobi or Gauss-seidal iterative method. (8)

$6x_1 - 2x_2 + x_3 = 11$
$-2x_1 + 7x_2 + 2x_3 = 5$
$X_1 + 2x_2 - 5x_3 = -1$

Solutions:

Rewriting the given equations, we get

$$x_1 = \frac{(11 + 2x_2 - x_3)}{6} \quad \dots\dots\dots (i)$$

$$x_2 = \frac{(5 + 2x_1 - 2x_3)}{7} \quad \dots\dots\dots (ii)$$

$$x_3 = \frac{(1 + x_1 - 2x_2)}{5} \quad \dots\dots\dots (iii)$$

If the initial approximation is $x_1^{(0)} = x_2^{(0)} = x_3^{(0)} = 0$ then from equation (i), (ii) and (iii), we get the first approximations as:

$$x_1^{(1)} = \frac{11}{6}$$

$$x_2^{(1)} = \frac{5}{7}$$

$$x_3^{(1)} = \frac{1}{5}$$

Now, for the second approximation, we have

$$x_1^{(2)} = \frac{\left(11 + 2 \times \frac{5}{7} - \frac{1}{5}\right)}{6} = 2.04$$

$$x_2^{(2)} = \frac{\left(5 + 2 \times \frac{11}{6} - 2 \times \frac{1}{5}\right)}{7} = 1.18$$

$$x_3^{(2)} = \frac{\left(1 + \frac{11}{6} + 2 \times \frac{5}{7}\right)}{5} = 0.85$$

Now, for the third approximation, we have

$$x_1^{(3)} = \frac{(11 + 2 \times 1.18 - 0.85)}{6} = 2.09$$

$$x_2^{(3)} = \frac{(5 + 2 \times 2.04 - 2 \times 0.85)}{7} = 1.05$$

$$x_3^{(3)} = \frac{(1 + 2.04 + 2 \times 1.18)}{5} = 1.08$$

Continuing the same way we get the following approximations:

$$x_1^{(4)} = 2.003 \quad ; \quad x_2^{(4)} = 1.003 \quad ; \quad x_3^{(4)} = 1.038$$

$$x_1^{(5)} = 1.995 \quad ; \quad x_2^{(5)} = 1.138 \quad ; \quad x_3^{(5)} = 1.002$$

$x_1^{(6)} = 2.046$; $x_2^{(6)} = 1.141$; $x_3^{(6)} = 1.054$

$x_1^{(7)} = 2.038$; $x_2^{(7)} = 1.148$; $x_3^{(7)} = 1.066$

$x_1^{(8)} = 2.038$; $x_2^{(8)} = 1.144$; $x_3^{(8)} = 1.067$

$x_1^{(9)} = 2.037$; $x_2^{(9)} = 1.144$; $x_3^{(9)} = 1.065$

We continue this process till we reach the required level of accuracy.

Q.N. 5) Write an algorithm & computer program to fit a curve $\boldsymbol{y = ax^2 + bx + c}$ for given sets of $(\boldsymbol{x_1}, \boldsymbol{y_1}, \boldsymbol{g}, \boldsymbol{0} = \boldsymbol{1}, \ldots . \boldsymbol{x})$ values by least square method. (4+8)

Let $(x_i, y_i), i = 1, 2, 3, \ldots, n$ be a given set of n pairs of data points and let $y = f(x)$ be the curve that is fitted to this data. At $x = x_i$, the given value of the ordinate is y_i and the corresponding value on the fitting curve is $f(x_i)$. Then the error of approximating at $x = x_i$ is

$$e_i = y_i - f(x_i)$$

Let

$$S = \left(y_1 - f(x_1)\right)^2 + \left(y_2 - f(x_2)\right)^2 + \cdots + \left(y_n - f(x_n)\right)^2 = e_1^2 + e_2^2 + \cdots + e_n^2 = \sum_{i=1}^{n} e_i^2$$

Then the method of least squares approximation consists of finding an expression $y = f(x)$ such that the sum of the squares of the errors is minimized. i.e. S is minimum.

Algorithm:

1. Input:
   Set of n data pairs $(x_i, y_i), i = 1, 2, 3, \ldots, n$
2. Process:
   SET $sum\ x = 0$, $sum\ y = 0$, $sum\ x_2 = 0$, $sum\ xy = 0$
   FOR $i = 1$ TO n
   { SET $sum\ x = sum\ x + x_i$
   SET $sum\ y = sum\ y + y_i$
   SET $sum\ x_2 = sum\ x_2 + x_i^2$
   SET $sum\ xy = sum\ xy + x_i y_i$
   }
   SET $b = \frac{n \times sum\ xy - sum\ x \times sum\ y}{n \times sum\ x_2 - sum\ x \times sum\ x}$

   SET $a = \frac{sum\ y - b \times sum\ x}{n}$

3. Output:
   The straight line of the equation $y = a + bx$

Program:

```
#include <stdlib.h>
#include <math.h>
#include <stdio.h>
int main(int argc, char **argv)
{       double *x, *y;
        double SUMx, SUMy, SUMxy, SUMxx, SUMres, res, slope, y_intercept, y_estimate;
        int i, n;
        FILE *infile;
        infile = fopen("xydata", "r");
        if (infile == NULL)
        printf("error opening file\n");
        fscanf (infile, "%d", &n);
        x = (double *) malloc (n*sizeof(double));
        y = (double *) malloc (n*sizeof(double));
        SUMx = 0;
        SUMy = 0;
        SUMxy = 0;
        SUMxx = 0;
        for (i=0; i<n; i++)
        {       fscanf (infile, "%lf %lf", &x[i], &y[i]);
                SUMx = SUMx + x[i];
                SUMy = SUMy + y[i];
                SUMxy = SUMxy + x[i]*y[i];
```

```
        SUMxx = SUMxx + x[i]*x[i];
    }
    slope = ( SUMx*SUMy - n*SUMxy ) / ( SUMx*SUMx - n*SUMxx );
    y_intercept = ( SUMy - slope*SUMx ) / n;
    printf ("\n");
    printf ("The linear equation that best fits the given data:\n");
    printf ("y = %6.2lfx + %6.2lf\n", slope, y_intercept);
    printf ("--------------------------------------------------\n");
    printf ("Original (x,y)     Estimated y     Residual\n");
    printf ("--------------------------------------------------\n");
    SUMres = 0;
    for (i=0; i<n; i++)
    {   y_estimate = slope*x[i] + y_intercept;
        res = y[i] - y_estimate;
        SUMres = SUMres + res*res;
        printf ("(%6.2lf %6.2lf)      %6.2lf %6.2lf\n",      x[i], y[i], y_estimate, res);
    }
    printf("--------------------------------------------------\n");
    printf("Residual sum = %6.2lf\n", SUMres);
    return 1;
}
```

Q.N.6)Derive a difference equation to represent Poisson's equation. Solve the Poisson's equation $\nabla^2 f = 2x^2y^2$ over the square to main $0 \le x \le 3, 0 \le y \le 3$ with $f = 0$ on the boundary and $h = 1$. **(3+5)**

<u>Difference equation to represent Poisson's equation:</u>

Let $u = u(x, y)$ be a function of two independent variables x & y. Then by Taylor's formula:

$$u(x+h, y) = u(x, y) + hu_x(x, y) + \frac{h^2}{2!}u_{xx}(x, y) + \frac{h^3}{3!}u_{xxx}(x, y) + \cdots \quad \ldots\ldots\ldots (i)$$

$$u(x-h,y) = u(x,y) - hu_x(x,y) + \frac{h^2}{2!}u_{xx}(x,y) - \frac{h^3}{3!}u_{xxx}(x,y) + \cdots \quad \dots\dots\dots(ii)$$

$$u(x,y+k) = u(x,y) + ku_y(x,y) + \frac{k^2}{2!}u_{yy}(x,y) + \frac{k^3}{3!}u_{yyy}(x,y) + \cdots \quad \dots\dots\dots(iii)$$

$$u(x,y-k) = u(x,y) - ku_y(x,y) + \frac{k^2}{2!}u_{yy}(x,y) - \frac{k^3}{3!}u_{yyy}(x,y) + \cdots \quad \dots\dots\dots(iv)$$

Adding equations (i) & (ii) and ignoring the terms containing h^4 and higher powers, we get

$$u(x+h,y) + u(x-h,y) = 2u(x,y) + h^2u_{xx}(x,y)$$

$$or, u_{xx}(x,y) = \frac{1}{h^2}[u(x+h,y) - 2u(x,y) + u(x-h,y)] \quad \dots\dots\dots(A)$$

Adding equations (iii) & (iv) and ignoring the terms containing k^4 and higher powers, we get

$$u(x,y+k) + u(x,y-k) = 2u(x,y) + k^2u_{yy}(x,y)$$

$$or, u_{yy}(x,y) = \frac{1}{k^2}[u(x,y+k) - 2u(x,y) + u(x,y-k)] \quad \dots\dots\dots(B)$$

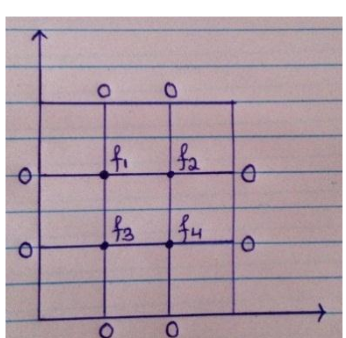
Now if $u_{xx} + u_{yy} = g(x,y)$ is the given Poisson's equation, then from equation (A) & (B) choosing $h = k$ we have,

$$u(x+h,y) + u(x,y+h) + u(x-h,y) + u(x,y-h) - 4u(x,y) = h^2g(x,y)$$

which is the difference equation for Poisson's equation.

Numerical:

The domain is divided as follows with $f = 0$ at the boundary



Now, from the difference equation for the Poisson's equation, we have

$$0 + 0 + f_2 + f_3 - 4f_1 = 1^2 \times 2 \times 1^2 \times 2^2$$

$$or, f_2 + f_3 - 4f_1 = 8 \quad \dots\dots\dots(i)$$

$0 + 0 + f_1 + f_4 - 4f_2 = 1^2 \times 2 \times 2^2 \times 2^2$

$or, f_1 + f_4 - 4f_2 = 32 \quad \dots\dots\dots (ii)$

$0 + 0 + f_1 + f_4 - 4f_3 = 1^2 \times 2 \times 1 \times 1$

$or, f_1 + f_4 - 4f_3 = 2 \quad \dots\dots\dots (iii)$

$0 + 0 + f_2 + f_3 - 4f_4 = 1^2 \times 2 \times 2^2 \times 1$

$or, f_2 + f_3 - 4f_4 = 8 \quad \dots\dots\dots (iv)$

Solving these equations, we get

$$f_1 = -\frac{11}{2}$$

$$f_2 = -\frac{43}{4}$$

$$f_3 = -\frac{13}{4}$$

$$f_4 = -\frac{11}{2}$$

Q.N.7) Define Order Differential Equation of the first order. What do you mean by initial value problem? Find by Taylor's series method, the values of y at $x = 0.1$ & $x = 0.2$ to fine places of decimal form.

$\frac{dy}{dx} = x^2y - 1$; $y(0) = 1$ **(2+6)**

An order differential equation (ODE) is an equation that contains one or several derivatives of an unknown function $y(x)$ having one independent variable x. Solving a differential equation means to find that unknown function $y(x)$ which satisfies the given differential equation. In that case, $y(x)$ is called the solution of the given differential equation. The order of differential equation is the order of the highest derivative that appears in the equation. $\frac{dy}{dx} = 3x^2 + y$ is a 1st order differential equation.

A general solution of a differential equation contains as many constants as the order of the differential equation. To eliminate more constants, we need a conditions on the solution of differential equation. When all the conditions are specified at a particular value of the independent variable x, then the problem is called an initial value problem and the conditions are called initial conditions.

Numerical:

The Taylor's series expansion of $y(x)$ at $x = 0$ is given by:

$$y(x) = y(0) + y'(0)(x-0) + \frac{y''(0)(x-0)^2}{2!} + \frac{y'''(0)(x-0)^3}{3!} + \frac{y''''(0)(x-0)^4}{4!} + \cdots$$

$$or, y(x) = y(0) + y'(0)x + \frac{y''(0)x^2}{2} + \frac{y'''(0)x^3}{3!} + \frac{y''''(0)x^4}{4!} + \cdots \quad \ldots\ldots\ldots (i)$$

Now,

$y(0) = 1$

$y'(0) = 0^2 \times 1 - 1 = -1$

$y''(x) = x^2y' + 2xy \Rightarrow y''(0) = 0^2y'(0) + 2 \times 0 \times y'(0) = 0$

$y'''(x) = x^2y'' + 4xy' + 2y \Rightarrow y'''(0) = 0 + 2y(0) = 2 \times 1 = 2$

$y''''(x) = x^2y''' + 6xy'' + 6y' \Rightarrow y''''(0) = 0 + 0 + 6y'(0) = 6 \times (-1) = -6$

Therefore, from equation (i), ignoring the terms containing x^5 and higher power, we get

$$y(x) = 1 - x + \frac{0}{2}x^2 + \frac{2}{6}x^3 - \frac{6}{24}x^4 = 1 - x + \frac{x^3}{3} - \frac{x^4}{4}$$

When $x = 0.1$, then

$$y(0.1) = 1 - 0.1 + \frac{0.1^3}{3} - \frac{0.1^4}{4} = 0.9 + 0.00033 - 0.000025 = 0.900305$$

When $x = 0.2$, then

$$y(0.2) = 1 - 0.2 + \frac{0.2^3}{3} - \frac{0.2^4}{4} = 0.8 + 0.002667 - 0.0004 = 0.802267$$