1. Compute with the following code

```
format long
table = zeros(6,6);
x = [0.15 0.21 0.23 0.27 0.32 0.35];
table(1:6,1) = F(x);

for q = 1:5
    for i = 1:6-q
        table(i,q+1) = (table(i+1,q) - table(i,q))/(x(i+q)-x(i));
    end

end

disp(table)
disp("row : xi, for i = 0 ~ 5")
disp("column : f[i j], for j = i ~ i+5\n")

for i = 1:4
    fprintf("from x%d:\n",i-1)
    fprintf("polynomial: %d + %d * (x-x%d) + %d * (x-x%d)(x-x%d)\n", table(i,1),table(i,2),i-1, table(i,3), i-1, i)
    fprintf("f'(x) = %d + %d * (x - x%d + x - x%d)\n", table(i,2), table(i,3), i-1, i)
    fprintf("f'(x) = %d\n\n", table(i,2) + table(i,3) * (0.268 - x(i) + 0.268 - x(i+1)))
end

fprintf("real answer: %d\n", F_pron(0.268))
fprintf("fitting with x2, x3, x4 will have the smallest error\n")
fprintf("this is intuitive because the sum of distance from x2, x3, x4 to 0.268 is the smallest")
function y = F(s)
    y = 1 + log10(s);
end

function y = F_pron(s)
    y = log10(exp(1))/s;
end
```

Result:

```
>> Q1
   0.176091259055681   2.435467261303968  -5.750502464003589  15.347400998804821 -38.811585996091971  94.579487309859431
   0.322219294733919   1.975427064183681  -3.908814344147010   8.749431379469184 -19.895688534120087                   0
   0.361727836017593   1.740898203534860  -2.946376892405400   5.964034984692373                   0                   0
   0.431363764158987   1.475724283218374  -2.230692694242316                   0                   0                   0
   0.505149978319906   1.297268867678989                   0                   0                   0                   0
   0.544068044350276                   0                   0                   0                   0                   0

row : xi, for i = 0 ~ 5
column : f[i j], for j = i ~ i+5\n
from x0:
polynomial: 1.760913e-01 + 2.435467e+00 * (x-x0) + -5.750502e+00 * (x-x0)(x-x1)
f'(x) = 2.435467e+00 + -5.750502e+00 * (x - x0 + x - x1)
f'(x) = 1.423379e+00

from x1:
polynomial: 3.222193e-01 + 1.975427e+00 * (x-x1) + -3.908814e+00 * (x-x1)(x-x2)
f'(x) = 1.975427e+00 + -3.908814e+00 * (x - x1 + x - x2)
f'(x) = 1.600181e+00

from x2:
polynomial: 3.617278e-01 + 1.740898e+00 * (x-x2) + -2.946377e+00 * (x-x2)(x-x3)
f'(x) = 1.740898e+00 + -2.946377e+00 * (x - x2 + x - x3)
f'(x) = 1.634829e+00

from x3:
polynomial: 4.313638e-01 + 1.475724e+00 * (x-x3) + -2.230693e+00 * (x-x3)(x-x4)
f'(x) = 1.475724e+00 + -2.230693e+00 * (x - x3 + x - x4)
f'(x) = 1.596182e+00

real answer: 1.620502e+00
fitting with x2, x3, x4 will have the smallest error
this is intuitive because the sum of distance from x2, x3, x4 to 0.268 is the smallest
```

2. Derivation process shown as following (b can be computed by deleting the last term in (a), the result of b is 1/h*(b+c*(2*s-1)/2))

(a)

Find $f'(x)$ ($a = table(1,1), b = \dots d = table(1,4)$)

$$f(x) = a + sb + \frac{s(s-1)}{2}c + \frac{s(s-1)(s-2)}{6}d \qquad \left(s = \frac{x-x_0}{h}\right)$$

$$f'(x) = \frac{ds}{dx} \cdot \frac{df}{ds} = \frac{1}{h} \times \left(b + \frac{2s-1}{2}c + \frac{d}{6}(s^2-3s+2+s^2-2s+s^2-s)\right)$$

$$= \frac{1}{h}\left(b + \frac{c(2s-1)}{2} + \frac{d(3s^2-6s+2)}{6}\right)$$

(c)

add another term on $f(x)$

$(s^2-s)(s^2-5s+6) = s^4-6s^3+11s^2-6s$

$$f(x) = a + sb + \frac{s(s-1)}{2}c + \frac{s(s-1)(s-2)}{6}d + \frac{s(s-1)(s-2)(s-3)}{24}e$$

$$f'(x) = \frac{1}{h}\left(b + \frac{c(2s-1)}{2} + \frac{d(3s^2-6s+2)}{6} + \frac{e(4s^3-18s^2+22s-6)}{24}\right)$$

Code:

```
table = zeros(7,7);
x = 0.3;
h = 0.2;
x_all = [x x+h x+2*h x+3*h x+4*h x+5*h x+6*h];
table(1:7,1) = f(x_all);
sign = 1;
for i = 2:7
    for j = 1:7-i+1
        sign = 1;
        for k = j+i-1:-1:j
            table(j,i) = table(j,i) + table(k,1) * nchoosek(i-1,k-j) * sign;
            sign = sign * -1;
        end
    end

end
disp(table)
disp(" (a)")
fprintf("   estimate using x1, x2, x3, x4\n")
fprintf("   s = (x - x0)/h = 2.1\n")
s = (0.72-0.5)/0.2;
fprintf("   f(x) = %d + s*%d + s*(s-1)*%d / 2 + s*(s-1)*(s-2)*%d / 6\n", table(2,1), table(2,2), table(2,3), table(2,4))
fprintf("   f'(x) = (%d + %d * (2*s - 1) / 2 + %d * (3*s^2 - 6*s + 2) / 6) / h\n", table(2,2), table(2,3), table(2,4))
fprintf("   f'(0.72) = %d\n\n", (table(2,2) + table(2,3) * (2*s - 1)/2 + table(2,4) * (3*s^2 - 6*s + 2)/6)/h)

disp(" (b)")
fprintf("   estimate using x4, x5, x6\n")
fprintf("   s = (x - x4)/h = 1.15\n")
s = (1.33-1.1)/0.2;
fprintf("   f(x) = %d + s*%d + s*(s-1)*%d / 2\n", table(5,1), table(5,2), table(5,3))
fprintf("   f'(x) = (%d + %d * (2*s - 1) / 2) / h\n", table(5,2), table(5,3))
fprintf("   f'(1.33) = %d\n\n", (table(5,2) + table(5,3) * (2*s - 1)/2)/h)

disp(" (c)")
fprintf("   estimate using x0, x1, x2, x3, x4\n")
fprintf("   s = (x - x0)/h = 1\n")
s = (0.5-0.3)/0.2;
fprintf("   f(x) = %d + s*%d + s*(s-1)*%d / 2 + s*(s-1)*(s-2)*%d / 6 + s*(s-1)*(s-2)*(s-3)*%d / 24\n", table(1,1), table(1,2), table(1,3), table(1,4), table(1,5))
fprintf("   f'(x) = (%d + %d * (2*s - 1) / 2 + %d * (3*s^2 - 6*s + 2) / 6 + %d * (4*s^3 - 18*s^2 + 22*s -6)/24) / h\n", table(1,2), table(1,3), table(1,4), table(1,5))
fprintf("   f'(0.72) = %d\n\n", (table(1,2) + table(1,3) * (2*s - 1)/2 + table(1,4) * (3*s^2 - 6*s + 2)/6 + table(1,5) * (4*s^3 - 18*s^2 + 22*s -6)/24)/h)


function y = f(x)
    y = x + sin(x) /3;
end
```

Result:

```
>> Q2
    0.3985    0.2613   -0.0064   -0.0022    0.0003    0.0001   -0.0000
    0.6598    0.2549   -0.0086   -0.0018    0.0004    0.0001         0
    0.9147    0.2464   -0.0104   -0.0014    0.0005         0         0
    1.1611    0.2360   -0.0118   -0.0010         0         0         0
    1.3971    0.2241   -0.0128         0         0         0         0
    1.6212    0.2113         0         0         0         0         0
    1.8325         0         0         0         0         0         0

 (a)
  estimate using x1, x2, x3, x4
  s = (x - x0)/h = 2.1
  f(x) = 6.598085e-01 + s*2.549307e-01 + s*(s-1)*-8.560975e-03 / 2 + s*(s-1)*(s-2)*-1.848615e-03 / 6
  f'(x) = (2.549307e-01 + -8.560975e-03 * (2*s - 1) / 2 + -1.848615e-03 * (3*s^2 - 6*s + 2) / 6) / h
  f'(0.72) = 1.250465e+00

 (b)
  estimate using x4, x5, x6
  s = (x - x4)/h = 1.15
  f(x) = 1.397069e+00 + s*2.241169e-01 + s*(s-1)*-1.280467e-02 / 2
  f'(x) = (2.241169e-01 + -1.280467e-02 * (2*s - 1) / 2) / h
  f'(1.33) = 1.078970e+00

 (c)
  estimate using x0, x1, x2, x3, x4
  s = (x - x0)/h = 1
  f(x) = 3.985067e-01 + s*2.613018e-01 + s*(s-1)*-6.371061e-03 / 2 + s*(s-1)*(s-2)*-2.189914e-03 / 6 + s*(s-1)*(s-2)*(s-3)*3.412991e-04 / 24
  f'(x) = (2.613018e-01 + -6.371061e-03 * (2*s - 1) / 2 + -2.189914e-03 * (3*s^2 - 6*s + 2) / 6 + 3.412991e-04 * (4*s^3 - 18*s^2 + 22*s -6)/24) / h
  f'(0.5) = 1.292548e+00
```

3. Finding undetermined coefficients

$$f''(x_0) = C_{-2}f_{-2} + C_{-1}f_{-1} + C_0 f_0 + C_1 f_1 + C_2 f_2$$

choose $P(u)=1,\ P(u)=u,\ P(u)=u^2,\ P(u)=u^3,\ P(u)=u^4$

case 1 : $P(u)=1$

$$f_{-2}=f_{-1}=\dots f_2=1$$

$$C_{-2}+C_{-1}+C_0+C_1+C_2=0 \quad \dots ①$$

case 2 : $P(u)=u$

$$f_{-2}=P(-2h)=-2h,\ f_{-1}=P(-h)=-h,\ f_0=P(0)=0,\ f_1=P(h)=h,\ f_2=P(2h)=2h$$

$$(-2h)C_{-2}+(-h)C_{-1}+0C_0+hC_1+2hC_2=0 \quad \dots ②$$

case 3 : $P(u)=u^2$

$$f_{-2}=4h^2,\ f_{-1}=h^2,\ f_0=0,\ f_1=h^2,\ f_2=4h^2$$

$$4h^2C_{-2}+h^2C_{-1}+0\cdot C_0+h^2C_1+4h^2C_2=\underbrace{P''(0)=2}_{\text{for } f''(x)} \quad \dots ③ \qquad \underbrace{=0}_{\text{for } f'''(x)}$$

case 4 : $P(u)=u^3$

$$(-8h^3)C_{-2}+(-h^3)C_{-1}+0C_0+h^3C_1+8h^3C_2=\underbrace{0}_{\text{for } f''(x)} \quad \dots ④ \qquad \underbrace{=6}_{\text{for } f'''(x)}$$

case 5 : $P(u)=u^4$

$$16h^4C_2+h^4C_{-1}+0C_0+h^4C_1+16h^4C_2=0 \quad \dots ⑤$$

$f''(x)$:

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ -2h & -h & 0 & h & 2h \\ 4h^2 & h^2 & 0 & h^2 & 4h^2 \\ -8h^3 & -h^3 & 0 & h^3 & 8h^3 \\ 16h^4 & h^4 & 0 & h^4 & 16h^4 \end{bmatrix} \begin{bmatrix} C_{-2} \\ C_{-1} \\ C_0 \\ C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \\ 0 \end{bmatrix}$$

$f'''(x)$:

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ -2h & -h & 0 & h & 2h \\ 4h^2 & h^2 & 0 & h^2 & 4h^2 \\ -8h^3 & -h^3 & 0 & h^3 & 8h^3 \\ 16h^4 & h^4 & 0 & h^4 & 16h^4 \end{bmatrix} \begin{bmatrix} C_{-2} \\ C_{-1} \\ C_0 \\ C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 6 \\ 0 \end{bmatrix}$$

$$f''(x) \approx \frac{-f_{-2}+16f_{-1}-30f_0+16f_1-f_2}{12h^2}$$

$$f'''(x) \approx \frac{-f_{-2}+2f_{-1}-2f_1+f_2}{2h^3}$$

The coefficients are computed by the following code:

```
syms h;
syms c1 c2 c3 c4 c5;
syms f1 f2 f3 f4 f5;
A = [1 1 1 1 1; -2*h -h 0 h 2*h; 4*h^2 h^2 0 h^2 4*h^2; -8*h^3 -h^3 0 h^3 8*h^3;16*h^4 h^4 0 h^4 16*h^4];

b = [0 0 2 0 0]';
x = [c1; c2; c3; c4; c5];
x = solve(A*x == b, x);
disp("for f''(x) : ")
disp(x)

b = [0 0 0 6 0]';
x = [c1;c2;c3;c4;c5];
x = solve(A*x == b, x);
disp("for f'''(x) : ")
disp(x)
```

Result:

```
for f''(x) :
    c1: -1/(12*h^2)
    c2: 4/(3*h^2)
    c3: -5/(2*h^2)
    c4: 4/(3*h^2)
    c5: -1/(12*h^2)

for f'''(x) :
    c1: -1/(2*h^3)
    c2: 1/h^3
    c3: 0
    c4: -1/h^3
    c5: 1/(2*h^3)
```

Formula and error terms of f’’(x) and f’’’(x):

For $f''(x) \approx \frac{-f_{-2}+16f_{-1}-30f_0+16f_1-f_2}{12h^2}$

using Taylor series

$$f_{-2} = f_0 - 2hf_0' + \frac{4h^2}{2}f_0'' - \frac{8h^3}{6}f_0''' + \frac{16h^4}{24}f_0^{(4)} - \frac{32}{120}h^5 f_0^{(5)} + \frac{64}{720}h^6 f_0^{(6)}$$

$$f_2 = f_0 + 2hf_0' + \frac{4h^2}{2}f_0'' + \frac{8h^3}{6}f_0''' + \frac{16h^4}{24}f_0^{(4)} + \frac{32}{120}h^5 f_0^{(5)} + \frac{64}{720}h^6 f_0^{(6)}$$

$$-f_{-2}-f_2 = -(2f_0 + 4h^2 f_0'' + \frac{4}{3}h^4 f_0^{(4)} + \frac{64}{360}h^6 f_0^{(6)}) \dots$$

$$f_{-1} = f_0 - hf_0' + \frac{h^2}{2}f_0'' - \frac{h^3}{6}f_0''' + \frac{h^4}{24}f_0^{(4)} - \frac{1}{120}h^5 f_0^{(5)} + \frac{1}{720}h^6 f_0^{(6)}$$

$$f_1 = f_0 + hf_0' + \frac{h^2}{2}f_0'' + \frac{h^3}{6}f_0''' + \frac{h^4}{24}f_0^{(4)} + \frac{1}{120}h^5 f_0^{(5)} + \frac{1}{720}h^6 f_0^{(6)}$$

$$f_{-1}+f_1 = 2f_0 + h^2 f_0'' + \frac{h^4}{12}f_0^{(4)} + \frac{1}{360}h^6 f_0^{(6)}$$

$$\frac{-f_{-2}+16f_{-1}-30f_0+16f_1-f_2}{12h^2} \approx \frac{\cancel{-2f_0} - 4h^2 f''(x_0) - \cancel{\frac{4}{3}h^4 f^{(4)}(x_0)} - \frac{64}{360}h^6 f_0^{(6)} \cancel{-30f_0} + \cancel{32f_0} + 16h^2 f''(x_0) + \cancel{\frac{4}{3}h^4 f^{(4)}(x_0)} + \frac{16}{360}h^6 f_0^{(6)}}{12h^2}$$

$$= \frac{12h^2 f''(x_0) - \frac{48}{360}h^6 f_0^{(6)}}{12h^2}$$

$$= f''(0) + \frac{h^4}{90}f^{(4)}(x_0)$$

error for $f''(x_0) = \frac{h^4}{90}f^{(4)}(\xi) = O(h^4)$ #

For $f'''(x) \approx \frac{-f_{-2}+2f_{-1}-2f_1+f_2}{2h^3}$

$$f_{-2} = f_0 - 2hf_0' + \frac{4h^2}{2}f_0'' - \frac{8h^3}{6}f_0''' + \frac{16h^4}{24}f_0^{(4)} - \frac{32h^5}{120}f_0^{(5)} + \frac{64h^6}{720}f_0^{(6)} \dots$$

$$f_2 = f_0 + 2hf_0' + \frac{4h^2}{2}f_0'' + \frac{8h^3}{6}f_0''' + \frac{16h^4}{24}f_0^{(4)} + \frac{32h^5}{120}f_0^{(5)} + \frac{64h^6}{720}f_0^{(6)} \dots$$

$$f_2 - f_{-2} = 4hf_0' + \frac{8h^3}{3}f_0''' + \frac{32h^5}{60}f_0^{(5)} \dots$$

$$f_{-1} = f_0 - hf_0' + \frac{h^2}{2}f_0'' - \frac{h^3}{6}f_0''' + \frac{h^4}{24}f_0^{(4)} - \frac{h^5}{120}f_0^{(5)} + \frac{h^6}{720}f_0^{(6)} \dots$$

$$f_1 = f_0 + hf_0' + \frac{h^2}{2}f_0'' + \frac{h^3}{6}f_0''' + \frac{h^4}{24}f_0^{(4)} + \frac{h^5}{120}f_0^{(5)} + \frac{h^6}{720}f_0^{(6)} \dots$$

$$-2(f_1 - f_{-1}) = -2(2hf_0' + \frac{h^3}{3}f_0''' + \frac{h^5}{60}f_0^{(5)}) = -4hf_0' - \frac{2h^3}{3}f_0''' - \frac{h^5}{30}f_0^{(5)} \dots$$

$$\frac{-f_{-2}+2f_{-1}-2f_1+f_2}{2h^3} \approx \frac{1}{2h^3} \times (2h^3 f_0''' + \frac{1}{2}h^5 f_0^{(5)}) = f_0''' + \frac{1}{4}h^2 f_0^{(5)}$$

error $= \frac{17}{60}h^2 f^{(5)}(\xi) = O(h^2)$ #

4. I interpolated the points using divided difference to get the estimated fuction and integrated the function to get the estemated integration result.
   Code:

```
format long
table = zeros(9,9);
x1 = [1.0 1.1 1.2 1.3 1.4 1.5 1.6 1.7 1.8];
table(1:9,1) = [1.543 1.669 1.811 1.971 2.151 2.352 2.577 2.858 3.107]';

for q = 1:8
    for i = 1:9-q
        table(i,q+1) = (table(i+1,q) - table(i,q))/(x1(i+q)-x1(i));
    end
end

f = @(x) table(1,1) + table(1,2)*(x-1) + table(1,3)*(x-1).*(x-1.1) + table(1,4)*(x-1).*(x-1.1).*(x-1.2) ...
+ table(1,5)*(x-1).*(x-1.1).*(x-1.2).*(x-1.3) + table(1,6)*(x-1).*(x-1.1).*(x-1.2).*(x-1.3).*(x-1.4) +  ...
table(1,7).*(x-1).*(x-1.1).*(x-1.2).*(x-1.3).*(x-1.4).*(x-1.5) + ...
table(1,8)*(x-1).*(x-1.1).*(x-1.2).*(x-1.3).*(x-1.4).*(x-1.5).*(x-1.6) + ...
table(1,9)*(x-1).*(x-1.1).*(x-1.2).*(x-1.3).*(x-1.4).*(x-1.5).*(x-1.6).*(x-1.7);

I = integral(f,1,1.8);
fprintf("   Estimated integral using interpolation = %d\n\n", I)

% do 1/3 first
sum = 1/30*(table(1,1)+4*table(2,1)+table(3,1)) + 3/80*(table(3,1)+3*table(4,1)+3*table(5,1)+table(6,1)) + 3/80*(table(6,1)+3*table(7,1)+3*table(8,1)+table(9,1));
fprintf("   using 1/3 first, result = %d\n",sum)
fprintf("   error = %d\n\n", abs(I-sum))

sum = 1/30*(table(4,1)+4*table(5,1)+table(6,1)) + 3/80*(table(1,1)+3*table(2,1)+3*table(3,1)+table(4,1)) + 3/80*(table(6,1)+3*table(7,1)+3*table(8,1)+table(9,1));
fprintf("   using 1/3 in the middle, result = %d\n",sum)
fprintf("   error = %d\n\n", abs(I-sum))

sum = 1/30*(table(7,1)+4*table(8,1)+table(9,1)) + 3/80*(table(1,1)+3*table(2,1)+3*table(3,1)+table(4,1)) + 3/80*(table(4,1)+3*table(5,1)+3*table(6,1)+table(7,1));
fprintf("   using 1/3 at the end, result = %d\n",sum)
fprintf("   error = %d\n\n", abs(I-sum))

fprintf("   using Simpson 3/8 for (x = 1~1.3 and x = 1.3~1.6), and using Simpson 1/3 for (x = 1.6~1.8) will have the smallest error\n")
```

   Result:

```
>> Q4
   Estimated integral using interpolation = 1.771864e+00

   using 1/3 first, result = 1.770321e+00
   error = 1.543153e-03

   using 1/3 in the middle, result = 1.770325e+00
   error = 1.538986e-03

   using 1/3 at the end, result = 1.770946e+00
   error = 9.181526e-04

   using Simpson 3/8 for (x = 1~1.3 and x = 1.3~1.6), and using Simpson 1/3 for (x = 1.6~1.8) will have the smallest error
```

   The interpolation result can be used to estimate the error of integration because it's error is way smaller than the error of integration

5. Compute with the following code:

```
format long
[answer, minh] = adaptive_trapezoidal(0.2,1);
minh = minh/2;
fprintf("   when terminate, h = %d\n", minh)

function [I, minh] = adaptive_trapezoidal(a, b)
    fa = f(a);
    fb = f(b);
    I = (b-a)/2 * (fa + fb);
    minh = b-a;
    intervalLeft = (b-a)/4 * (fa + f((a+b)/2));              % integrate h/2 with left
    intervalRight = (b-a)/4 * (f((a+b)/2) + fb);             % integrate h/2 with right
    if abs(I - (intervalLeft + intervalRight)) > 0.02       % if "next" iteration have error more than 0.02
        [integralLeft, hleft] = adaptive_trapezoidal(a, (a+b)/2);
        [integralRight, hright] = adaptive_trapezoidal((a+b)/2, b);
        I = integralRight + integralLeft;
        minh = min(hleft, hright);
    end
end

function y = f(x)
    y = 1/x^2;
end
```

   the recursion ends when the result of this and next recursion < 0.02, and the h is the b-a of this iteration, so the result of minh should be devided by 2.

   Result:

```
>> Q5
   when terminate, h = 2.500000e-02
```

6. Compute with the following code

```
disp(compute_area(-0.2, 1.4, 0.4, 2.6))


function I = compute_area(xleft, xright, yleft, yright)
    t = [-0.77459667 0 0.77459667];
    w = [0.55555555 0.88888889 0.55555555];
    newy = ((yright-yleft)*t+yright+yleft)/2;
    newx = ((xright-xleft)*t+xright+xleft)/2;
    factor = (xright-xleft)*(yright-yleft)/4;
    I = 0;
    for i = 1:3
        for j = 1:3
            I = I + w(i)*w(j)*f(newx(i), newy(j));
        end
    end
    I = I * factor;
end

function z = f(x,y)
    z = exp(x)*sin(2*y);
end
```

Result:

```
>> Q6
   0.372377716481867
```