```matlab
%Preperation
h = 1 ./ 2.^(5+[1:5]);

sp2 = @(x) (x<0).*0 + ...
    (x>= 0 & x < 1).* (1/2*x.^2) + ...
    (x>= 1 & x < 2).* (3/4 - (x-3/2).^2) + ...
    (x>= 2 & x < 3).* (1/2*(x-3).^2);
sp3 = @(x) (x<0).*0 + ...
    + (x>= 0 & x < 1).* (1/6 * x.^3)  + ...
    + (x>= 1 & x < 2).* (1/6 * (1 + 3*(x-1) + 3*(x-1).^2 - 3*(x-1).^3)) + ...
    + (x>= 2 & x < 3).* (1/6 * (1 + 3*(3-x) + 3*(3-x).^2 - 3*(3-x).^3)) + ...
    + (x>= 3 & x < 4).* (1/6 * (4-x).^3);
dsp2 = @(x)  (x<0).*0 + ...
    (x>= 0 & x < 1).* x + ...
    (x>= 1 & x < 2).* (3-2*x) + ...
    (x>= 2 & x < 3).* (x-3);
dsp3 = @(x) (x<0).*0 + ...
    (x>= 0 & x < 1).* (1/2 * x.^2) + ...
    (x>= 1 & x < 2).* (1/2 * (1+2*(x-1)-3*(x-1).^2)) + ...
    (x>= 2 & x < 3).* (1/2 * (-1 - 2*(3-x) + 3*(3-x).^2)) + ...
    (x>= 3 & x < 4).* (- 1/2 * (4-x).^2);

x = @(h) [h:h:3-h];
y = @(h) [h:h:4-h];

hp_norm = @(x, h, p) (h .* (sum(abs(x).^p))).^(1/p);
hinf_norm = @(x) (max(abs(x)));

% Start
run_a = true;
run_b = true;
run_c = true;
run_d = true;
```

```
if(run_a)
    R_sp2 = @(h) 1/h * (sp2(x(h)) - sp2(x(h)-h));
    F = @(h) dsp2(x(h)) - R_sp2(h);

    p = 1;

    tmp = [];
    for i = 1:4
        tmp(i) = hp_norm(F(h(i)), h(i), p) ./ hp_norm(F(h(i+1)),h(i+1),p);
    end
    w = log2(mean(tmp));
    fprintf("a) w%i = %.1f\n" , p, w);
end

if(run_b)
    Z_sp2 = @(h) 1/2/h * (sp2(x(h) + h) - sp2(x(h)-h));
    F = @(h) dsp2(x(h)) - Z_sp2(h);

    tmp = [];
    for i = 1:4
        tmp(i) = hinf_norm(F(h(i))) ./ hinf_norm(F(h(i+1)));
    end
    w_inf = log2(mean(tmp));
    fprintf("b) w_inf = %.1f\n" , w_inf);
end
```

```matlab
    if(run_c)
        V_dsp3 = @(h) 1/h * (sp3(y(h) + h) - sp3(y(h)));
        F = @(h) dsp3(y(h)) - V_dsp3(h);

        p = 2;

        tmp = [];
        for i = 1:4
            tmp(i) = hp_norm(F(h(i)), h(i), p) ./ hp_norm(F(h(i+1)),h(i+1),p);
        end
        w = log2(mean(tmp));
        fprintf("c) w%i = %.1f\n" , p, w);
    end

    if(run_d)
        Z_dsp3 = @(h) 1/2/h * (sp3(y(h) + h) - sp3(y(h)-h));
        F = @(h) dsp3(y(h)) - Z_dsp3(h);

        p = 1;

        tmp = [];
        for i = 1:4
            tmp(i) = hp_norm(F(h(i)), h(i), p) ./ hp_norm(F(h(i+1)),h(i+1),p);
        end
        w = log2(mean(tmp));
        fprintf("d) w%i = %.1f\n" , p, w);
    end
```

Command Window

```
a) w1 = 1.0
b) w_inf = 1.0
c) w2 = 1.0
d) w1 = 2.0
>>
```

**Bsp 4** In einer Umgebung $B_x = B(x, \varepsilon)$ mit einem fixierten $\varepsilon > 0$ sei eine Funktion $f$ für die folgenden finiten Differenzen mit $0 < h \ll \varepsilon$ definiert. Kreuze bei den wahren Behauptungen an:

**a)** Für $f \in C^4(B_x)$ gilt:

$$f^{(2)}(x) = \underbrace{\frac{f(x) - 2f(x+h) + f(x+2h)}{h^2}}_{*} - h f^{(3)}(x) + \mathcal{O}(h^2)$$

Mit Taylor:

$$f(x+h) \approx f(x) + hf'(x) + h^2 \frac{f^{(2)}(x)}{2} + h^3 \frac{f^{(3)}(x)}{6} + \mathcal{O}(h^4)$$

$$f(x+2h) \approx f(x) + 2hf'(x) + 4h^2 \frac{f^{(2)}(x)}{2} + 8h^3 \frac{f^{(3)}(x)}{6} + \mathcal{O}(h^4)$$

einsetzen in rechte Seite ergibt:

$$* = \frac{f(x) - 2\left(f(x) + hf'(x) + h^2 f^{(2)}(x)/2 + h^3 f^{(3)}(x)/6 + \mathcal{O}(h^4)\right)}{h^2} +$$

$$+ \frac{f(x) + 2hf'(x) + 4h^2 f^{(2)}(x)/2 + 8h^3 f^{(3)}(x)/6 + \mathcal{O}(h^4)}{h^2}$$

$$= f^{(2)}(x) + hf^{(3)}(x) + \mathcal{O}(h^2)$$

[$\mathcal{O}(h^2) + \mathcal{O}(h^2) = \mathcal{O}(h^2)$]
$c\,\mathcal{O}(h^2) = \mathcal{O}(h^2)$

✓

**=> a) ist richtig**

**b)** Für $f \in C^6(B_x)$ gilt:

$$f^{(3)}(x) = \underbrace{\frac{f(x) - 3f(x-h) + 3f(x-2h) - f(x-3h)}{h^3}}_{\circledast} + \frac{3h}{2} f^{(4)}(x) + \mathcal{O}(h^3)$$

$$f(x-h) \approx f(x) - hf'(x) + h^2 \frac{f^{(2)}(x)}{2} - \frac{h^3 f^{(3)}(x)}{6} + \frac{h^4 f^{(4)}(x)}{24} - \frac{h^5 f^{(5)}(x)}{120} + \mathcal{O}(h^6)$$

$$f(x-2h) \approx f(x) - 2hf'(x) + 4h^2 f^{(2)}(x)/2 - 8h^3 f^{(3)}(x)/6 + 16h^4 f^{(4)}(x)/24 - \frac{32h^5 f^{(5)}(x)}{120} + \mathcal{O}(h^6)$$

$$f(x-3h) \approx f(x) - 3hf'(x) + 9h^2 \frac{f^{(2)}(x)}{2} - 27h^3 \frac{f^{(3)}(x)}{6} + 81h^4 \frac{f^{(4)}(x)}{24} - 243 \frac{h^5 f^{(5)}(x)}{120} + \mathcal{O}(h^6)$$

in ($\circledast$) einsetzen:

$$\frac{f(x) - 3\left(f(x) - hf'(x) + h^2 f^{(2)}(x)\cdot\frac{1}{2} - h^3 f^{(3)}(x)\frac{1}{6} + h^4 f^{(4)}(x)\cdot\frac{1}{24} - h^5 f^{(5)}(x)\cdot\frac{1}{120} + \mathcal{O}(h^6)\right)}{h^3} +$$

$$+ \frac{3\left(f(x) - 2hf'(x) + 4h^2 f^{(2)}(x)\frac{1}{2} - 8h^3 f^{(3)}(x)\frac{1}{6} + 16h^4 f^{(4)}(x)\frac{1}{24} - 32h^5 f^{(5)}(x)\frac{1}{120} + \mathcal{O}(h^6)\right)}{h^3}$$

$$- \frac{f(x) + 3hf'(x) - 9h^2 f^{(2)}(x)\frac{1}{2} + 27h^3 f^{(3)}(x)\frac{1}{6} - 81h^4 f^{(4)}(x)\frac{1}{24} + 243h^5 f^{(5)}(x)\frac{1}{120} + \mathcal{O}(h^6)}{h^3}$$

$$= f^{(3)}(x) - \frac{3h}{2} f^{(4)}(x) + \frac{h^2}{25} f^{(5)}(x) + \mathcal{O}(h^3)$$

Wenn man Taylor für $f \in C^5$ entwickelt erhält man:

$$\circledast = f^{(3)}(x) - \frac{3h}{2} f^{(4)}(x) + \mathcal{O}(h^2)$$

**=> b) ist falsch**

c) Für $f \in C^7(B_x)$ gilt:

$$f^{(2)}(x) = \underbrace{\frac{35 f(x) - 104 f(x+h) + 114 f(x+2h) - 56 f(x+3h) + 11 f(x+4h)}{12h^2}}_{(*)} + \frac{5h^3}{6} f^{(5)}(x) + O(h^5)$$

$$f(x+h) \approx f(x) + h f'(x) + h^2 \frac{f^{(2)}(x)}{2} + h^3 \frac{f^{(3)}(x)}{6} + h^4 \frac{f^{(4)}(x)}{24} + h^5 \frac{f^{(5)}(x)}{120} + h^6 \frac{f^{(6)}(x)}{720} + O(h^7)$$

analog für $f(x+2h)$ & $f(x+3h)$

in (*) einsetzen:

$$= \frac{35 f(x) - 104 \left( f(x) + h f'(x) + h^2 f^{(2)}(x) \frac{1}{2} + h^3 f^{(3)}(x) \frac{1}{6} + h^4 f^{(4)}(x) \frac{1}{24} + h^5 f^{(5)}(x) \frac{1}{120} + \right.}{12h^2}$$

$$\frac{+ h^6 f^{(6)}(x) \frac{1}{720} + O(h^7) \Big) + 114 \Big( f(x) + 2h f'(x) + 4h^2 f^{(2)}(x) \frac{1}{2} + 8h^3 f^{(3)}(x) \frac{1}{6} +}{12h^2}$$

$$\frac{+ 16h^4 f^{(4)}(x) \frac{1}{24} + 32 h^5 f^{(5)}(x) \frac{1}{120} + 64 h^6 f^{(6)}(x) \frac{1}{720} + O(h^7) \Big) -}{12h^2}$$

$$\frac{- 56 \Big( f(x) + 3h f'(x) + 9h^2 f^{(2)}(x) \frac{1}{2} + 27h^3 f^{(3)}(x) \frac{1}{6} + 81 h^4 f^{(4)}(x) \frac{1}{24} + 243 h^5 f^{(5)}(x) \frac{1}{120} +}{12h^2}$$

$$\frac{+ 729 h^6 f^{(6)}(x) \frac{1}{720} + O(h^7) \Big) + 11 \Big( f(x) + 4h f'(x) + 16h^2 f''(x) \frac{1}{2} + 64h^3 f^{(3)}(x) \frac{1}{6} + 256h^4 f^{(4)}(x) \frac{1}{24} +}{12h^2}$$

$$\frac{+ 1024 h^5 f^{(5)}(x) \frac{1}{120} + 4096 h^6 f^{(6)}(x) \frac{1}{720} + O(h^7) \Big)}{12h^2}$$

$$= f^{(2)}(x) + \frac{5h^3}{6} f^{(5)}(x) + \frac{119}{90} h^4 f^{(6)}(x) + O(h^5)$$

$\Rightarrow$ c) ist nicht richtig

Wenn man $f \in C^4$ wählt würde man erhalten:

$$(*) = f^{(2)}(x) + \frac{5}{6} h^3 f^{(5)}(x) + O(h^4)$$

d) Für $f \in C^5(B_x)$ gilt:

$$f'(x) = \frac{11 f(x) - 18 f(x-h) + 9 f(x-2h) - 2 f(x-3h)}{6h} + \frac{h^3}{4} f^{(4)}(x) + O(h^4)$$

$$f(x-h) \approx f(x) - h f'(x) + h^2 \frac{f^{(2)}(x)}{2} - h^3 \frac{f^{(3)}(x)}{6} + h^4 \frac{f^{(4)}(x)}{24} + O(h^5)$$

analog $f(x-2h)$, $f(x-3h)$
in rechte Seite einsetzen:

$$= \frac{11 f(x) - 18 (f(x) - h f'(x) + h^2 f^{(2)}(x) \frac{1}{2} - h^3 f^{(3)}(x) \frac{1}{6} + h^4 f^{(4)}(x) \frac{1}{24} + O(h^5))}{6h} +$$

$$+ \frac{9 (f(x) - 2h f'(x) + 4h^2 f^{(2)}(x) \frac{1}{2} - 8h^3 f^{(3)}(x) \frac{1}{6} + 16h^4 f^{(4)}(x) \frac{1}{24} + O(h^5))}{6h} -$$

$$- \frac{2 (f(x) - 3h f'(x) + 9h^2 f^{(2)}(x) \frac{1}{2} - 27h^3 f^{(3)}(x) \frac{1}{6} + 81h^4 f^{(4)}(x) \frac{1}{24} + O(h^5))}{6h}$$

$$= f'(x) - \frac{h^3}{4} f^{(4)}(x) + O(h^4)$$

$\Rightarrow$ d) ist richtig ✓

| Christoph Raunjak | Blatt 9 | Numerik I SS20 |
|---|---|---|

**Aufgabe 6**

☐ a.) ☑ b.) ☐ c.) ☑ d.)

Implementierung einer Funktion, die Werte für $f(x), g(x)$ und die benötigten exakten Ableitungen liefert. Auch eine symbolische Implementierung wäre möglich.

```
%*****************************************************************************
%*                                                                           *
%*                        func.m                                             *
%*                                                                           *
%*  Calculates function value / value of derivative                          *
%*                                                                           *
%*  Param:  x.. value at x                                                   *
%*          name.. name of the function, f,g possible                        *
%*          n.. nth derivative 0,2,4 possible                                *
%*  Out:    y.. value of name^(n)(x)                                         *
%*                                                                           *
%*  Author: Christoph Raunjak                                                *
%*          christoph.raunjak@edu.uni-graz.at                                *
%*  3.12.2020 vs1.0                                                          *
%*****************************************************************************
```

```
function y = func(x, name, n)
  switch name
    case 'f'
      switch n
        case 0
          y = 1 ./ (1+x); %f
        case 2
           y = 2 ./ (x+1).^3; %f^(2)
        case 4
          y = 24 ./ (x+1).^5; %f^(4)
        otherwise
          y = 0;
          disp('[ERR] unknown derivation!')
      endswitch
    case 'g'
      if mod(n,2) == 0
        y = exp(-x);     %g=g^(2)=g^(2k)
      else
        y = 0;
        disp('[ERR] unknown derivation!')
      endif
    otherwise
      y = 0;
      disp('[ERR] unknown function!)')
  endswitch
endfunction
```

Eine Implementierung der gegebenen Differenzenoperatoren für $f(x), g(x)$:

```
%**************************************************************************
%*                                                                        *
%*                          D_op.m                                        *
%*                                                                        *
%*   Calculates the derivate with the given approximation formula        *
%*                                                                        *
%*   Param:   x.. calculate D_hf^(n) at (x)                               *
%*            h.. h                                                       *
%*            fname.. function for which Df^(n) gets calculated           *
%*            n.. calculate the nth derivative                            *
%*   Out:     d.. D_hf^(n)(x)                                             *
%*                                                                        *
%*   Author: Christoph Raunjak                                            *
%*           christoph.raunjak@edu.uni-graz.at                            *
%*   3.12.2020 vs1.0                                                      *
%**************************************************************************

function d = D_op(x, h, fname, n)
  if n == 2
    d = (1/h^2) * (2*func(x,fname,0) - 5*func(x+h,fname,0)...
    + 4*func(x+2*h,fname,0) - func(x+3*h,fname,0));
  elseif n == 4
    d = (1/h^4) * (func(x,fname,0)-4*func(x+h,fname,0)+6*func(x+2*h,fname,0)...
    - 4*func(x+3*h,fname,0) + func(x+4*h,fname,0));
  else
  d = 0;
  disp('[ERR] Wrong parameter n!')
  endif
endfunction
```

Implementierung der Fehlernorm aus dem Skriptum

```
%**************************************************************************
%*                                                                        *
%*                          fehlernorm.m                                  *
%*                                                                        *
%*   Calculates the error norm from script page 195                       *
%*   derivative of certain function.                                      *
%*                                                                        *
%*   Param:   x.. a vector to calculate the norm of                       *
%*            p.. using the regular p-norm implemented in Matlab          *
%*            h.. the weight h used in the norm                           *
%*   Out:     n.. the error norm of x                                     *
%*                                                                        *
%*   Author: Christoph Raunjak                                            *
%*           christoph.raunjak@edu.uni-graz.at                            *
%*   3.12.2020 vs1.0                                                      *
%**************************************************************************

function n = fehlernorm(x, p, h)
  switch p
    case Inf
      n = norm(x,Inf);
    otherwise
      n = norm(x, p) * (h^(1/p));
  endswitch
endfunction
```

Implementierung einer Funktion um $\omega_p$ zu bestimmen (Zeilen 29-32) dienen nur zu späteren Visualisierung.

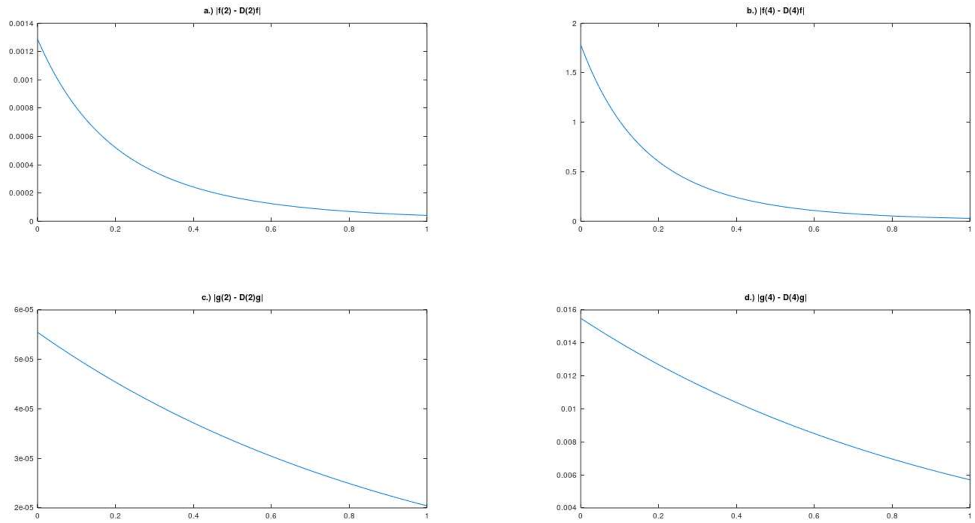
```
%**************************************************************************
%*                                                                        *
%*                        omega.m                                         *
%*                                                                        *
%*  Calculates \omega_p                                                   *
%*                                                                        *
%*  Param:  fname.. function for which \omega gets calculated             *
%*          n.. calculate \omega_p for the nth derivative                 *
%*          p.. which p should be used in fehlernorm.m                    *
%*  Out:    w.. \omega_p                                                  *
%*          DF.. f^(n)(x) for x_h with h = h_5                            *
%*          DF_approx.. f^(n)(x) approximated with given formula for x_h_5 *
%*                                                                        *
%*  Author: Christoph Raunjak                                             *
%*          christoph.raunjak@edu.uni-graz.at                             *
%*  3.12.2020 vs1.0                                                       *
%**************************************************************************
```

```
function [w, DF, DF_approx] = omega(fname, n, p)
  H = [];
  for i = 1:5
    H = [H, 1/(2^(2+i))];
  endfor
  F_h = [];
  for i = 1:5
    x = (0:H(i):1);
    F = func(x, fname, n) - D_op(x,H(i),fname,n);
    F_h = [F_h, fehlernorm(F,p,H(i))];
    if i == 5
      DF = func(x, fname, n);
      DF_approx = D_op(x,H(i),fname,n);
    endif
  endfor
  ratio = [];
  for i = 1:4
    ratio = [ratio, F_h(i) / F_h(i+1)];
  endfor

  w = log2(mean(ratio));

endfunction
```

Ergebnis der zu berechnenden $\omega_p$

```
a.) Answer = 1.915
b.) Answer = 0.896 (correct)
c.) Answer = 1.998
d.) Answer = 0.956 (correct)
```

Visualisierung der Fehler:



a.) |f(2) - D(2)f|
0.0014
0.0012
0.001
0.0008
0.0006
0.0004
0.0002
0
0 0.2 0.4 0.6 0.8 1
b.) |f(4) - D(4)f|
2
1.5
1
0.5
0
0 0.2 0.4 0.6 0.8 1
c.) |g(2) - D(2)g|
6e-05
5e-05
4e-05
3e-05
2e-05
0 0.2 0.4 0.6 0.8 1
d.) |g(4) - D(4)g|
0.016
0.014
0.012
0.01
0.008
0.006
0.004
0 0.2 0.4 0.6 0.8 1